NASA Contractor Report 166096

NASA-CR-166096
19840014159

Review and Verification of CARE III Mathematical Model and Code: Interim Report

D. M. Rose, R. E. Altschul, J. W. Manke, and D. L. Nelson

BOEING COMPUTER SERVICES COMPANY
SEATTLE, WASHINGTON 98124

Contract NAS1-16900
April 1983

National Aeronautics and Space Administration

Langley Research Center
Hampton, Virginia 23665

This Page Intentionally Left Blank

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

This Page Intentionally Left Blank

LIST OF FIGURES

This Page Intentionally Left Blank

LIST OF TABLES

This Page Intentionally Left Blank

LIST OF SYMBOLS

SYMBOL	DEFINITION
A	Active coverage state.
A_D	Active detected coverage state.
A_E	Active error coverage state.
$A'(t\|\underline{\ell})$	System failure rate due to critical pairs.
$a(t\|x)$	Probability that a given stage-x module has a latent fault at time t, given it has experienced a fault.
$a_i(t)$	Table 4.2-3.
$a'(t\|\underline{\ell})$	System failure rate due to error propagation.
B	Benign coverage state.
B_D	Benign detected coverage state.
B_E	Benign error coverage state.
$B(t,(x,y)\|\underline{\ell})$	Expected number of times a given (x,y) pair is latent in-use and critical at time t, given $\underline{\ell}$ faults.
$b_i(t)$	Table 4.2-3.
$b^{(1)}_{x,y}(\underline{\mu},\underline{\ell})$	Probability that a given (x,y) pair is critical when chosen from existing latent in-use stage-x and stage-y modules, given $\underline{\mu}$ latent and $\underline{\ell}$ faulty modules.
$b^{(2)}_{x,y}(\underline{\mu},\underline{\ell})$	Probability that a given (x,y) pair of modules is critical when chosen from existing latent in-use stage-x modules, and fault-free in-use stage-y modules, given $\underline{\mu}$ latent and $\underline{\ell}$ faulty modules.
C	Probability that a propagated error is detected before it causes the system to malfunction.
$c(x,a)$	Coverage status of module (x,a).
$\underline{c}$	Vector of components c(x,a).
$c(t\|y,\underline{\ell})$	Probability that a new fault on a fault-free in-use stage-y module would produce a system failure at time t, given $\underline{\ell}$ faults at time t-0.

LIST OF SYMBOLS (Continued)

SYMBOL	DEFINITION
$c_i(t)$	Table 4.2-4.
D	Detected coverage state.
DP	Detected as permanent (non-transient) coverage state.
DF	Double fault failure coverage state.
$D(t,(x,y)\|\underline{\ell})$	Expected number of (x,y) critical faults that would be created at time t as a result of a fault on a fault-free in-use stage-y module, given $\underline{\ell}$ faults at time t-0.
d(t)	Survival function for probability density function $\delta(t)$.
d(x,a)	Indicator of faulty status of module (x,a)
$\underline{d}$	Vector of components d(x,a).
E	Error coverage state.
e(t)	Survival function for probability density function e(t).
F	Failure coverage state.
$F(\underline{\ell})$	Aggregate of failure states with $\underline{\ell}$ faults.
f(t)	Table 4.2-1.
$f_i(t)$	Table 4.2-4.
$G(\underline{\ell})$	Aggregate of operational states with $\underline{\ell}$ faults.
$H(\underline{\ell})$	Aggregate of spares exhaustion states with $\underline{\ell}$ faults.
$H_B^-(t\|x_i)$	Probability that a given stage-x module has a non-benign category x_i fault at time t.
$H_B^-(t\|x)$	Probability that a given stage-x module has a non-benign fault at time t.
$H_B(t\|x_i)$	Probability that a given stage-x module has a benign category x_i fault at time t.

LIST OF SYMBOLS (Continued)

SYMBOL	DEFINITION
$H_L(t\|x_i)$	Probability that a given stage-x module has a latent category x_i fault at time t.
$H_L(t\|x)$	Probability that a given stage-x module has a latent fault at time t.
$h_{DF}(t\|x_i,y_j)$	Rate at which an (x_i,y_j) critical pair causes system failure.
$h_F(t\|x_i)$	Error propagation failure rate due to category x_i fault.
$i(x,a)$	Fault category index for module (x,a).
$\underline{i}$	Vector of components $i(x,a)$.
$\ell(x)$	Number of faulty modules in stage-x.
$\underline{\ell}$	Vector of components $\ell(x)$.
$\underline{\ell}+\underline{1}(x)$	$(\ell(1), \ell(2),..., \ell(x) + 1,...)$.
$\mathscr{L}$	Set of all possible fault vectors $\underline{\ell}$.
L	Set of $\underline{\ell}$ states which do not cause system failure by exhaustion.
$\bar{L}$	Spares exhaustion failure set of $\underline{\ell}$ states.
M	Total number of modules in the system.
m(x)	Minimum number of modules necessary for stage-x to be operational.
N	Number of stages in the system.
N(x,y)	Number of (x,y) critical pairs.
NOP	Spares schedule parameter matrix.
n(x)	Number of modules in stage-x.
P_A	Probability that a fault detected in the active state is diagnosed as permanent.

LIST OF SYMBOLS (Continued)

SYMBOL	DEFINITION
P_B	Probability that a fault detected in the benign state is diagnosed as permanent.
$P_B(t)$	Table 4.2-2.
$P_{\bar{B}}(t)$	Table 4.2-2.
$P_{DP}(t)$	Table 4.2-2.
$P_L(t)$	Table 4.2-2.
$P_a(t)$	Table 4.2-2.
$P_b(t)$	Table 4.2-2.
$P_{dp}(t)$	Table 4.2-2.
$P_e(t)$	Table 4.2-2.
$P(t\|\underline{\ell})$	Probability that the system is in state $G(\underline{\ell})$ at time t.
$P^*(t\|\underline{\ell})$	Probability that the system is in state $\underline{\ell}$ at time t, given perfect coverage.
$P(\mu(x), t\|\ell(x)$	Probability of $\mu(x)$ stage-x latent faults, given $\ell(x)$ faults.
$p_{DF}(t)$	Table 4.2-4.
$p_e(t)$	Table 4.2-2.
$p_{\bar{e}}(t)$	Table 4.2-2.
$p_F(t)$	Table 4.2-2.
$p_f(t)$	Table 4.2-2.
$p_3(t)$	Table 4.2-4.
$Q(t\|\underline{\ell})$	Probability that the system is in state $F(\underline{\ell})$ at time t.
$q(x)$	Number of in-use stage-x modules.

LIST OF SYMBOLS (Continued)

SYMBOL	DEFINITION
$R(t)$	Reliability of the system at time t.
$r(t)$	Survival function for probability density function $\rho(t)$.
$r(t\|x)$	Reliability of a stage-x module.
$S(t\|\underline{\ell})$	Probability that the system is in state $H(\underline{\ell})$ at time t.
x	Stage index.
(x,a)	a-th module in stage-x.
x_j	j-th fault category for stage-x.
Y_t	State of the system at time t.
α	Transition rate from active to benign
β	Transition rate from benign to active.
$\delta(t)$	Probability density function for transition from active to detected.
$\epsilon(t)$	Probability density function for transition out of error state.
$\Lambda(\mu,t\|\underline{\ell})$	Definite integral of $\lambda(t\|\underline{\ell})$.
$\Lambda^*(\mu,t\|\underline{\ell})$	Definite integral of $\lambda^*(t\|\underline{\ell})$.
$\lambda(t\|x_j)$	Rate of occurrence of fault category x_j.
$\lambda^{(1)}(t\|\underline{\ell},\underline{\ell}+\underline{1}(y))$	Rate for transition from $G(\underline{\ell})$ to $G(\underline{\ell}+\underline{1}(y))$.
$\lambda^{(2)}(t\|\underline{\ell},\underline{\ell}+\underline{1}(y))$	Rate for transition from $G(\underline{\ell})$ to $F(\underline{\ell}+\underline{1}(y))$.
$\lambda(t\|\underline{\ell})$	Transition rate out of $G(\underline{\ell})$.
$\lambda^*(t\|\underline{\ell})$	Transition rate out of $\underline{\ell}$ under perfect coverage.
$\lambda^*(t\|\underline{\ell},\underline{\ell}+\underline{1}(y))$	Rate for transition from $H(\underline{\ell})$ to $H(\underline{\ell}+\underline{1}(y))$ or from $G(\underline{\ell})$ to $H(\underline{\ell}+\underline{1}(y))$ or from $\underline{\ell}$ to $\underline{\ell}+\underline{1}(y)$ under perfect coverage.

LIST OF SYMBOLS (Continued)

SYMBOL	DEFINITION
$\mu(x)$	Number of latent modules in stage-x.
$\underline{\mu}$	Vector of components $\mu(x)$.
$\mu(t\|\underline{\ell})$	Rate for transition from G($\underline{\ell}$) to F($\underline{\ell}$).
$\rho(t)$	Probability density function for transition from active to error.
$\phi(t)$	Table 4.2-2.
$\chi_B(t)$	Table 4.2-2.
$\psi_A(t)$	Table 4.2-2.
$\psi_B(t)$	Table 4.2-2.

Section 1
INTRODUCTION AND SUMMARY

This Interim Report documents the CARE-III mathematical model and code verification performed by Boeing Computer Services from January 1982 through November 1982. The mathematical model has been verified for permanent and intermittent faults. The transient fault model has not been addressed. The code verification has been performed on CARE-III, Version 3. A CARE-III Version 4, which corrects deficiencies identified in Version 3, is being developed as part of the overall study.

1.1 BACKGROUND OF PROBLEM

Fault-tolerant flight control systems (FTFCS) are designed to be ultra-reliable. Key modules are redundant to a level that makes the probability of failure due to spares exhaustion extremely small. These systems are designed to mask the faulty operation of a failed module until the system can successfully reconfigure with a spare. This masking of the faulty module to an observer outside the system comprises the fault-tolerance of the system. System failure due to improper masking is called a coverage failure. These systems are designed with sufficient redundancy such that coverage failure greatly dominates spares exhaustion as the mode of system failure.

The reliability of any proposed FTFCS must obviously be demonstrated. Since the reliability needs to be extremely high, assessment of the reliability must come from engineering analysis and reliability modeling. Laboratory testing of a system with mean time between failure greater than 10^6 hours is obviously not practical.

For each proposed FTFCS, a reliability model and program could conceivably be developed. The alternative is to develop a general reliability program which permits the representation of systems with diverse architecture and

fault-masking techniques. Under NASA funding, Raytheon has been developing such a program. CARE-III represents the current level of this sequential development.

1.2 CARE-III GENERAL APPROACH

CARE-III is a reliability program which permits the evaluation of complex redundant fault-tolerant systems. The program was designed for the evaluation of FTFCS but is sufficiently general to permit, in principle, use for a wide variety of systems. This generality is discussed in more detail in Section 2.2.

The reliability model which CARE-III addresses is a semi-Markov process with an unmanageably large number of states. Assumptions about the relative size of the module failure and coverage parameters permit this detailed micro model to be approximated by a macro Markov model with a greatly reduced number of states. Furthermore, replacing detailed information contained in the micro model by probabilities of the corresponding events in the macro model permits the separation of the reliability model into a coverage model, which must be solved only once, and a reliability program which uses the coverage model output (see Section 2.3).

1.3 OBJECTIVE OF THIS PROJECT

The objectives of this project are the verification of the mathematical model and the computer code (Task 1) and the test stressing (Task 2) of CARE-III. This interim report addresses the results to date on Task 1. Additional Task 1 results, and all Task 2 findings will be addressed in the final report.

Task 1

In the mathematical model verification, equations are to be independently derived from the basic model. The solution approach implemented is to be

investigated with respect to accuracy and stability. Approximations used in the simplification of the model and the solution approach are to be reviewed and evaluated.

In the computer code verification the program structure, algorithms and equations are to be reviewed. In the program structure review modularity, maintainability, internal structure logic and data storage are to be evaluated. The choice and implementation of algorithms, for numerical solving or evaluating equations, are to be reviewed. Equations derived from the code are compared to those from the mathematical model.

This verification process is intended to assure that CARE-III is a mathematically valid reliability tool for a well defined set of problems.

1.4 QUESTIONS TO BE ADDRESSED

During this investigation of CARE-III, a number of questions have arisen. Many of these have been resolved and are addressed in this interim report.

- Documentation - The original theory document, Phase II, for CARE-III was inadequate for describing the model and program. This interim report is intended to fill some of that void.

- Mathematical Model

  - The detailed stochastic model for the system represented
  - The simplified stochastic model approximating the detailed model
  - The derivation of transition rates
  - The solution approach (differential equation versus integral equation versus approximate integral equation solutions)
  - Coverage
  - Transient failure model

- Model Implemented in Code

  - Program architecture
  - Algorithms chosen
  - Approximations in solution implementation
  - Program efficiency
  - Reliability

    - System architecture for computing Q and P*
    - Sparing algorithm

- Representation of FTFCS - A user's guide is needed which shows how the user may go from an understanding of a system (system structure, error rates, detection isolation and reconfiguration rates) to representing the system in CARE-III.

  - What systems may be represented by CARE-III?
  - How is software failure modeled?
  - How are stage and module dependencies handled?

1.5 SUMMARY OF RESULTS

1.5.1 General Comments

During Task 1, Model Verification, the BCS team has extensively reviewed the CARE-III model, the CARE-III documentation and the CARE-III program (Version 3, 1982). We agree with a vast majority of the material evaluated. There are, however, several areas which we feel need either further development or reworking.

The development of the CARE-III program shows an understanding of the basic requirements for a reliability program for a FTFCS. The basic structure chosen, a non-homogeneous semi-Markov process, appears to provide a general structure which permits representing the operation of a

FTFCS (e.g., scheduled computations; sparing; majority voters; permanent, intermittent and transient faults; fault coverage). Due to the excessively large number of states in this model for any reasonable system, implementation is not feasible. Dr. Stiffler replaces this model with a much simpler approximation, a non-homogeneous Markov model with a greatly reduced number of states. Appropriate assumptions on relative transition rates permit the separate solution of the coverage and reliability models. The solution of the resultant model is feasible, and is implemented in the CARE-III program. The CARE-III code exhibits good program structure and organization. Comments within the program highlight the calculations performed in the various subroutines. For a reasonably large and complex program, over 4500 lines of FORTRAN code, relatively few coding errors were identified during the review.

1.5.2 Documentation

The incomplete existing documentation for CARE-III poses a major problem for anyone interested in investigating or understanding the model and using the code. The underlying theoretical model, the solution approach, the implementation of the model into code, and the choice of algorithms implemented are not well documented in the Raytheon Phase II reports (Stiffler, J. J. and Bryant, L. A. (1982); Bryant, L. A. and Stiffler, J. J. (1982 a,b)). The intent of this report is to fill some of this void.

For a FTFCS designer, the existing User's Manual, Bryant, L. A. and Stiffler, J. J. (1982b), does not provide sufficient guidance in the use of CARE-III. The principal problem of representation, transferring system design information into input parameters, is not addressed. The meaning and use of several input parameters are inadequately described. In order to make CARE-III a useful tool, this shortcoming must be remedied.

1.5.3 Theoretical Model

The reliability model which CARE-III implements is an approximation to the detailed, but intractible, reliability model which better represents a FTFCS. The assumptions necessary for, and the limitations as a result of this approximation are not detailed in the CARE-III documentation. In the derivation of rates within the model, we take exception with several of the formulas. The b_{xy}, as defined in the CARE-III documentation, and as implemented in code, lead to some questionable results. These terms, necessary for computing double-fault coverage probabilities, give different answers if one numbers modules from left to right or right to left. We believe the definition of b_{xy} needs to be changed and have provided a solution (equations 3.4-11 and 3.4-22).

The transient case appears to pose a problem for CARE-III. The most natural way to represent transient faults is through a reversible model. That is, transitions from ℓ to ℓ - 1 are possible. An irreversible model is, however, much easier to solve. CARE-III, an irreversible model, addresses transients by modifying the fault occurrence rate. This approach has led to computational problems in the code. We believe that these problems can be avoided by calculating the intensity of entry into the detected as permanent state instead of its probability. It should be noted that the formulas used for transient faults in the derivation of rates have not been validated and require further investigation.

1.5.4 Model Implementation and Code

The reliability model implemented in CARE-III is defined by a system of ordinary differential equations for the probabilities of the system to be in operational states (P's), coverage failure states (Q's) and exhaustion failure states (S's). The differential equations are not solved directly by a numerical integration method, but rather the integral solution of the equations is computed using numerical quadrature methods. BCS suggests that this decision should be reconsidered. To compute the solution, the

P's in the integral equations for Q and S are replaced by the perfect coverage probabilities (P^*'s), which can be computed directly. The impact of this approximation on the estimation of the system reliability is not addressed in the CARE-III documentation or monitored in the program.

The calculation of system reliability is partitioned into "SUBRUN's" which consist of the evaluation of the reliability of subsystems which are independent in the sense that modules in different subsystems are not critically coupled. The calculation of system reliability from SUBRUN results appears to be in error and is under current investigation.

1.5.5 Algorithms and Data Structures

The solution of the CARE-III reliability model requires the implementation of algorithms for the numerical integration of a function and the numerical convolution of two functions. The quadrature rule used for numerical integration is Simpson's Rule, and it is adequately programmed in CARE-III. However, the stepsize for the integration is proportional to the flight time, since the array sizes for the reliability functions are fixed. This may degrade the accuracy of the solution for long flight times. The method of moments is used for numerical convolution of the module failure rate functions with the coverage failure rate functions. The implementation is based on the assumption that the coverage failure rate functions decay quickly to zero in the time scale of the module failure rate functions. BCS questions whether this assumption is valid in all cases; the resolution of the question is important because the convolution is the vital link between the coverage and reliability models.

The solution of the CARE-III coverage model requires the implementation of algorithms for the numerical sum of two functions, the numerical integration of a function, the numerical convolution of two functions and the numerical solution of Volterra integral equations of the second kind. The procedure for computing the numerical sum, Simpson's rule for numerical integration and the Trapezoidal rule for numerical convolution are ade-

quately programmed in CARE-III. The procedure for solving Volterra integral equations is based on linear interpolation and the numerical convolution algorithm. Although the procedure is adequately programmed in CARE-III, BCS has questioned its numerical stability; this is a subject of current study.

All the algorithms for the coverage model are closely tied to a CARE-III data structure, which permits only doublings of the discrete stepsize. This is based on the expectation that all coverage functions are exponentially decaying, positive functions. The heuristics in the numerical sum, convolution and Volterra algorithms indicate that not all coverage functions meet the expectation. More general algorithms and data structures may improve the computations in the coverage model.

Section 2
OVERVIEW OF CARE-III MODEL

The reliability model implemented in CARE-III is designed to assess the reliability of complex, redundant, fault-tolerant systems such as FTFCS's; requirements for the model are briefly discussed in Section 2.1. As pointed out in Section 2.2, the detailed model used in CARE-III requires the solution of a semi-Markov process with an excessively large number of states. An aggregation procedure, discussed in Section 2.3, is used in CARE-III to reduce the solution task to the solution of an aggregate reliability model and a coverage model.

The aggregate reliability model is briefly reviewed in Section 2.4 below and is then discussed in detail in Section 3; the theory and derivation of the model is given in Sections 3.1 to 3.4 and its implementation in CARE-III is described in Sections 3.5 to 3.6. The coverage model is briefly reviewed in Section 2.5 below and is then discussed in detail in Section 4; the theory and derivation of the model is given in Sections 4.1 to 4.2 and its implementation in CARE-III is described in Sections 4.3 to 4.4.

2.1 REQUIREMENTS FOR MODELING FTFCS's

In Phase I of the research conducted by the Raytheon Company, several FTFCS's were studied to determine requirements for the CARE-III reliability model; these included SIFT, FTMP, ARCS and FTSC. The requirements established by the Raytheon researchers, J. J. Stiffler, L. A. Bryant, L. Guccione (1979) pp. 12-16, are quoted below:

- "Capability of modeling up to at least 40 stages"
- "Multiple operating modes for each set of coupled stages"
- "Separate coverage model similar to that in CARE-II but capable of handling latent and intermittent faults as well as permanent faults"
- "Multiple success criteria"
- "N-point failure mechanisms"
- "Time-dependent hazard rates"
- "Transient faults"
- "Non-unity dormancy factors"

2.2 FORMULATION OF CARE-III MODEL

In the usual reliability analysis of a system, the system is operational if at least a specified set of modules is operational. In a FTFCS, the system is operational if at least a specified set of modules is operational, and the faulty modules have not caused a faulty operation of the system before they are detected, isolated, and replaced. The masking of the faulty operation of a module before it is replaced is called fault coverage and is part of the fault tolerance of the system. For a FTFCS the reliability of the system is the probability that at least a specified set of modules is operational and that all faulty modules have been covered.

2.2.1 Modules and Stages

The basic units represented in CARE-III are modules (e.g., processor, memory, bus); groups of identical modules are called stages. A stage is operational if at least a specified number of modules in that stage are operational.

For stage x, define

$n(x)$ = number of stage x modules,
$m(x)$ = minimum number of x modules necessary for stage operation,
$\ell(x)$ = number of failed stage x modules.

Then, assuming perfect coverage and independence of modules, the probability of stage x being operational is a sum of binomial probabilities (see Section 3.5.4).

2.2.2 Exhaustion

The system is composed of N independent stages. Assuming perfect coverage, the system is in either an operational or a failed state based

on the state of each of the N stages and the system logic. Module status is used only in determining the state of the stage.

System failure caused by stage failure(s) is termed failure by exhaustion. This is best illustrated by example. Consider a system composed of two stages, x and y. This system is in an operational state if either x or y are operational. Then the reliability R of the system, using the independence of stages, is

$$
\begin{aligned}
R &= P(\text{system is operational}) \\
&= 1 - P(\text{system failed}) \\
&= 1 - P(\text{stage x failed and stage y failed}) \\
&= 1 - P(\text{stage x failed}) \cdot P(\text{stage y failed})
\end{aligned}
$$

2.2.3 Coverage

For a FTFCS, the operational status of the stages, together with system architecture is insufficient to determine if the system is operational. The status of each failed module must be known, together with some system logic, to determine whether a failed module, or a pair of failed modules, have caused propagation of an error and system failure before the failed modules are detected, isolated, and replaced by spare modules. System failure due to improper error masking is termed a coverage failure.

In the ultra-reliable FTFCS imagined for future aircraft, coverage failure is presumed to be the dominant mode of failure. To address this failure mode a detailed model is necessary to carry the relevant information on the status of the failed modules.

A common fault tolerant technique to mask errors is to use triplexes with majority voting. Three identical modules perform the same task. If a single module is faulty, the voter should mask the error. Failure of the voter to mask the error to the system is a single fault coverage (system)

failure. If two modules in the triplex are faulty, the majority voter can't mask the error. These modules are said to be critically coupled. There may be spare stage modules available, but the existence of two faulty modules in the triplex brings the system down before the faulty modules may be replaced. This is a double fault coverage (system) failure.

The coverage problem for a module is represented by a multi-state semi-Markov model discussed in Sections 2.5, 3.1 and 4.2. The coverage model addresses both single and double fault failure. Higher order failure combinations are not addressed by CARE III. In a pentaplex system with majority, 3-out-of-5, voting critical triples need to be represented.

2.2.4 Fault Categories

A module may be susceptible to more than one mode of failure, each mode having its own occurrence rate. These competing risks on the module may have different coverage parameters. This level of detail is included in the CARE-III model and discussed in Sections 2.5, 3.1 and 4.1.

2.2.5 State Space

Given the system, stage, and coverage structure, the state of the system is determined by the status of the stages and the faulty modules. The module information must include which modules have failed, fault category, and coverage state. Let (x,a) denote the a-th module in stage x. The fault information for (x,a) is completely specified as below by the vector d(x,a), i(x,a), c(x,a), up to sparing. A representation of sparing could be accomplished by the inclusion of an additional indicator variable.

d(x,a) = 0 if (x,a) operational,
1 if (x,a) faulty,

i(x,a) = Fault category (0 to 5),

$c(x,a)$ = Coverage state of module (A, B,...).

Let M denote the number of modules in the system, then

$$M = \sum_{x=1}^{N} n(x),$$

and the state of the system is completely specified by the three M dimensional vectors:

$$\underline{d} = (d(1,1),\ldots, d(N,n(N)),$$
$$\underline{i} = (i(1,1),\ldots, i(N,n(N)),$$
$$\underline{c} = (c(1,1),\ldots, c(N,n(N)).$$

If there are only two stages (N = 2), with three modules per stage (n(1) = n(2) = 3), three fault types, and five coverage states, the number of ($\underline{d}$, $\underline{i}$, $\underline{c}$) states is

$$7.29 \times 10^8 = 2^6 \cdot 3^6 \cdot 5^6.$$

Not all of these states are possible. If d = 0, then i = c = 0. This reduces the number of states to

$$1.68 \times 10^7 = (3 \cdot 5 + 1)^6.$$

This number can be further reduced, since not all these stages are possible; yet the number of states remaining is still huge. Given the failure rates for the fault types, the coverage parameters, and the system, stage and coverage structure, the stochastic model is fully specified. However, solving the resulting integral equations for state probabilities for even this simple system is not feasible. For any system of moderate size and complexity, N will be much larger, as will be n(x), and the number of states increases exponentially. Some solution approach is required which reduces the dimensionality of the problem.

2.3 SOLUTION OF CARE-III MODEL

An approximate solution of the detailed reliability model for CARE-III is obtained by solving a reduced order reliability model that is constructed by an aggregation procedure. The states of the system are grouped into aggregate states defined by the following system data:

- Number of faulty modules in each stage,
- System fault tree,
- Coverage structure,
- Critical pairs fault trees.

Rates for transitions between aggregate states are defined as aggregates of the rates for transitions between detailed states in the aggregate states. Approximate values for these rates are defined by an averaging procedure that decomposes the coverage and reliability calculations. The resulting reduced order reliability model is still a semi-Markov process; however, it is approximately a Markov process under the assumption that:

- The coverage rates are much greater than the module failure rates.

The aggregation procedure thus decomposes the solution of the detailed reliability model into the solution of two problems of lower dimension:

- Coverage Model;
  - semi-Markov-process, and

- Reliability Model;
  - non-homogenous, Markov-process.

These models are discussed briefly in Sections 2.4 and 2.5 below and then in detail in Sections 3.1 - 3.4 and 4.1 - 4.2; the rest of this section outlines the construction of the aggregate state space.

The aggregate states are indexed by "fault vectors" which specify the number of faulty modules in each stage:

$$\mathcal{L} = \left\{ \underline{\ell} = (\ell(1), \ell(2), \ldots \ell(N)) : 0 \leq \ell(x) \leq n(x),\ 1 \leq x \leq N \right\}$$

For a particular fault vector, stage x has failed by spares exhaustion if $\ell(x) > n(x) - m(x)$; the system has failed by spares exhaustion if the system fault tree specifies that the set of failed stages for $\underline{\ell}$ is a system failure. This decomposes the set of fault vectors into two sets:

$$\mathcal{L} = L \cup \overline{L},$$

where the system is operational for fault vectors in L and failed for fault vectors in $\overline{L}$.

The aggregate states shown in Figure 2.3-1 are defined as follows:

- $\underline{\ell} \in \overline{L}$;

$$H(\underline{\ell}) = \left\{ (\underline{d},\underline{i},\underline{c}) : \sum_a d(x,a) = \ell(x),\ 1 \leq x \leq N \right\},$$

- $\underline{\ell} \in L$;

$$G(\underline{\ell}) = \left\{ \begin{array}{l} (\underline{d},\underline{i},\underline{c}) : \sum_a d(x,a) = \ell(x),\ 1 \leq x \leq N\ , \\ \text{and } \underline{c} \text{ does not specify any single or double fault} \\ \text{failures.} \end{array} \right\},$$

$$F(\underline{\ell}) = \left\{ \begin{array}{l} (\underline{d},\underline{i},\underline{c}) : \sum_a d(x,a) = \ell(x),\ 1 \leq x \leq N\ , \\ \text{and } \underline{c} \text{ specifies at least a single or double fault} \\ \text{failure.} \end{array} \right\}.$$

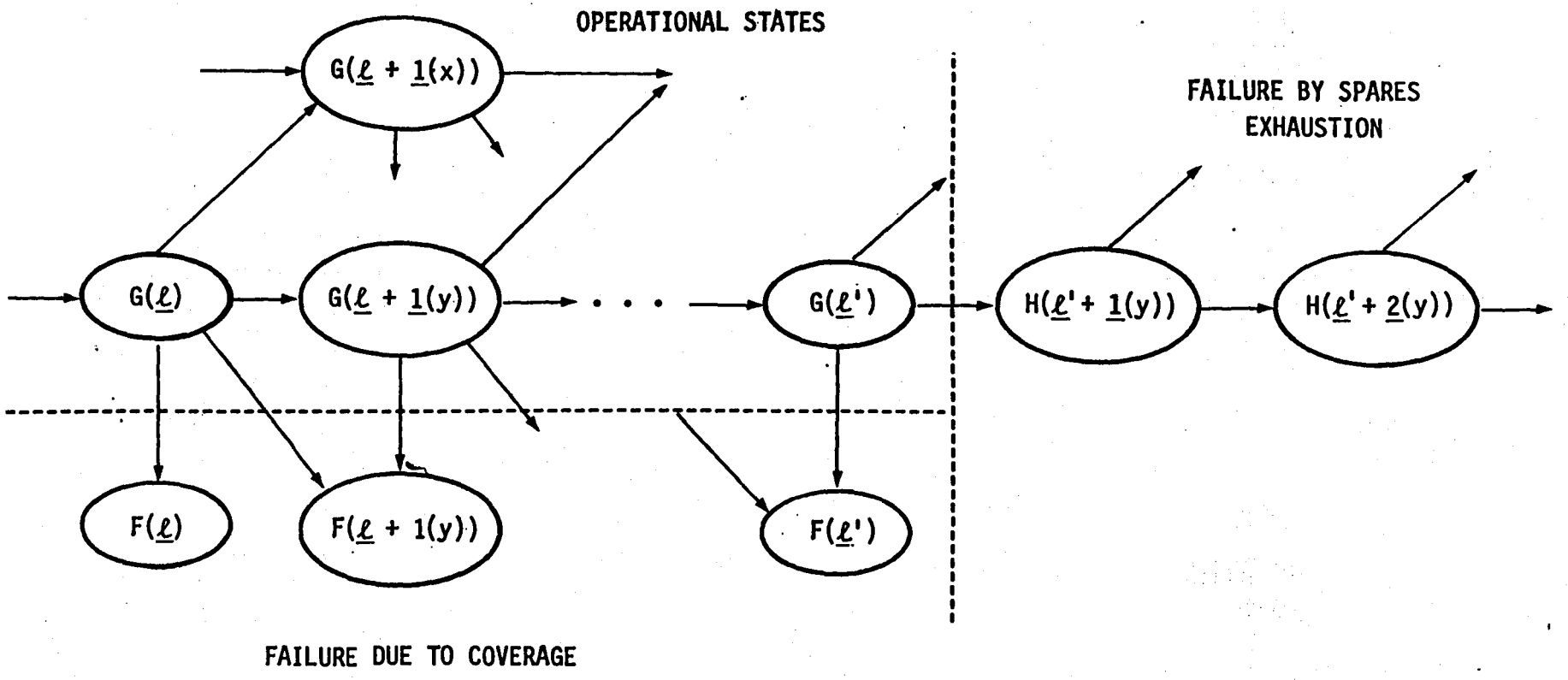


Figure 2.3-1 General Structure of CARE III Aggregate Model

The criteria for including ($\underline{d},\underline{i},\underline{c}$) in $G(\underline{\ell})$ or $F(\underline{\ell})$ based on $\underline{c}$ is explained in detail in Sections 3.1.3-4 where the single and double fault coverage models are described.

The possible transitions between the aggregate states are illustrated in Figure 2.3-2; $\underline{\ell}$ is an arbitrary fault vector and $\underline{\ell} + \underline{1}(y)$ differs from $\underline{\ell}$ only in having one more fault in stage-y:

- Case (a) : $G(\underline{\ell})$ to $F(\underline{\ell})$
  In case (a), $\underline{\ell} \in L$ and the only transition is from $G(\underline{\ell})$ to $F(\underline{\ell})$ due to single fault coverage failure.

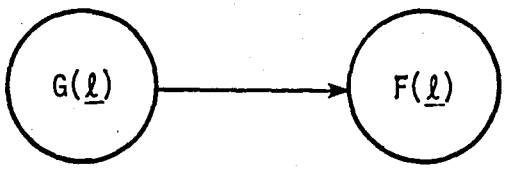
- Case (b) : $G(\underline{\ell})$ to $G(\underline{\ell}+1(y))$ or $F(\underline{\ell}+\underline{1}(y))$
  In case (b), $\underline{\ell}$ and $\underline{\ell} + \underline{1}(y) \in L$ and the possible transitions are from $G(\underline{\ell})$ to $G(\underline{\ell}+\underline{1}(y))$ or from $G(\underline{\ell})$ to $F(\underline{\ell}+\underline{1}(y))$ due to a double fault coverage failure.

- Case (c) : $G(\underline{\ell})$ to $H(\underline{\ell}+\underline{1}(y))$
  In case (c), $\underline{\ell} \in L$ and $\underline{\ell} + \underline{1}(y) \in \bar{L}$ and the only transition is from $G(\underline{\ell})$ to $H(\underline{\ell}+\underline{1}(y))$ due to system failure by exhaustion.

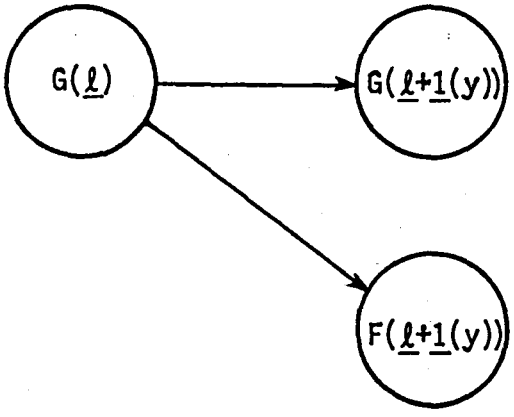
- Case (d) : $H(\underline{\ell})$ to $H(\underline{\ell}+\underline{1}(y))$
  In case (d), $\underline{\ell}$ and $\underline{\ell} + \underline{1}(y) \in \bar{L}$ and the only transition is from $H(\underline{\ell})$ to $H(\underline{\ell}+\underline{1}(y))$. These transitions are included in the model since the $H(\underline{\ell})$ states are not treated as absorbing states.

No transitions from the $F(\underline{\ell})$ states are defined in the model since the $F(\underline{\ell})$ states are treated as absorbing states.

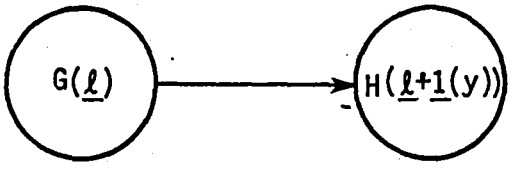
A complete description of the aggregate reliability model is given in Section 3.2 and the calculation of the rates for transitions between aggregate states is presented in Section 3.4. The discussion illustrates how the aggregation procedure decomposes the solution of the high order detailed reliability model for CARE-III into the solution of the low order coverage and aggregate reliability models.



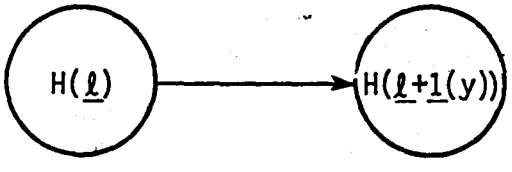
Case (a): $\underline{\ell} \in L$



Case (b): $\underline{\ell}$, $\ell + \underline{1}(y) \in L$



Case (c): $\underline{\ell} \in L$, $\underline{\ell} + \underline{1}(y) \in \bar{L}$



Case (d): $\underline{\ell}$, $\underline{\ell} + 1(y) \in \bar{L}$

Figure 2.3-2 Transition Between Aggregate States

2.4 RELIABILITY MODEL

The reliability of the system at time t, R(t), is given by

$$R(t) = P \text{ (system in state } G(\underline{\ell}) \text{ at time } t, \ \underline{\ell} \in L)$$

and is computed by

$$\begin{aligned} R(t) &= 1 - P \text{ (system in state } F(\underline{\ell}) \text{ at } t, \ \underline{\ell} \in L) - \\ &\qquad P \text{ (system in state } H(\underline{\ell}) \text{ at } t, \ \underline{\ell} \in L) \\ &= 1 - \sum_{\underline{\ell} \in L} P(F(\underline{\ell}) \text{ at } t) - \sum_{\underline{\ell} \in \bar{L}} P(H(\underline{\ell}) \text{ at } t) \\ &= 1 - \sum_{\underline{\ell} \in L} Q(t|\underline{\ell}) - \sum_{\underline{\ell} \in \bar{L}} S(t|\underline{\ell}) \end{aligned}$$

A system fault tree specifies the system structure in terms of the stages. These failure paths, together with the vector $(\underline{n(x) - m(x)})$, determine the sets L and $\bar{L}$ for which the probabilities $Q(t \mid \underline{\ell})$ and $S(t \mid \underline{\ell})$ must be computed (see Sections 2.3 and 3.5.2).

Consider a system with two stages. Let

$$\begin{aligned} n(1) &= n(2) = 3 \\ m(1) &= m(2) = 2 \end{aligned}$$

Then each stage may experience a single fault and still operate. If the stages are in series, as described by the system fault tree, the $\ell(1)$, $\ell(2)$ space is partitioned with respect to Q and S computation as shown in Figure 2.4-1.

If the stages are in parallel, the $\ell(1)$, $\ell(2)$ space is partitioned as shown in Figure 2.4-2.

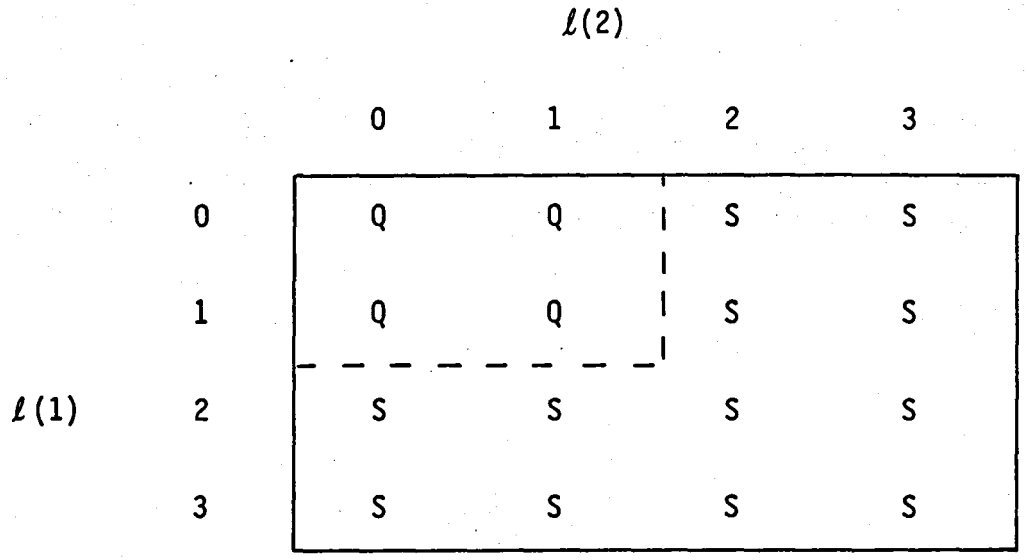


Figure 2.4-1 $\underline{\ell}$ space partition for series system

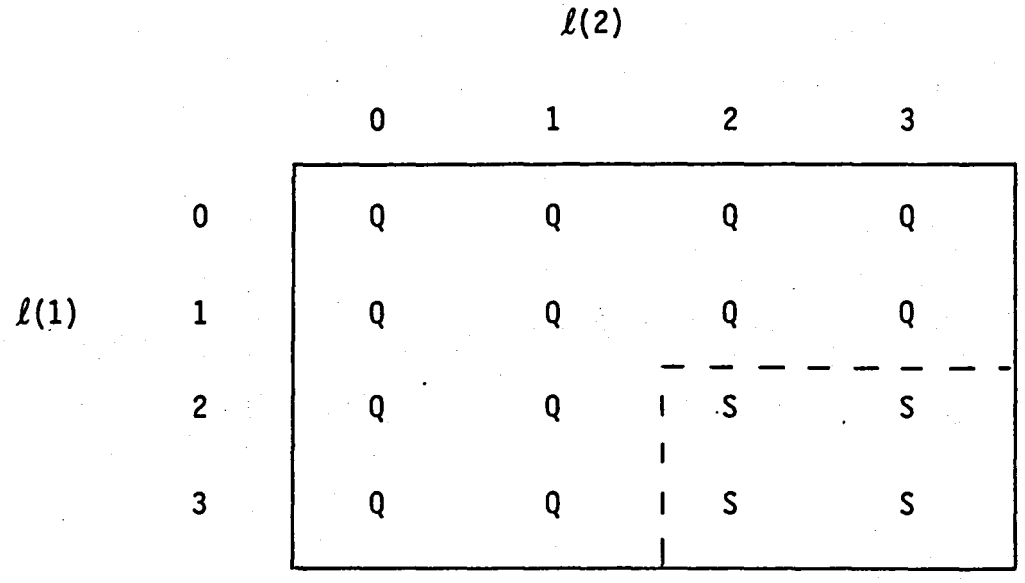


Figure 2.4-2 $\underline{\ell}$ space partition for parallel system

To obtain $S(t|\underline{\ell})$ and $Q(t|\underline{\ell})$ one must solve the integral equations since the process is semi-Markov. The approximation of this process by a Markov process, as discussed in Sections 2.3 and 3.2.4 permits one to obtain $S(t|\underline{\ell})$ and $Q(t|\underline{\ell})$ from the differential equations (3.2-1 to 3.2-6). Solution of these differential equations by direct numerical integration was rejected by the CARE-III developers after considering a simple difference equation. (Other numerical integration procedures are available and should be considered.) The differential equations are solved in CARE-III by using numerical quadrature methods to evaluate the formal integral solution of the equations.

CARE-III does not solve the differential equations for the $S(t|\underline{\ell})$. Under the assumption that the coverage transition rates are much larger than the failure rates, perfect coverage probabilities $P^*(t|\underline{\ell})$ are solved for (equation 3.3-1), where $P^*(t|\underline{\ell})$ is just the product of the N binomial probabilities for stage x to have $\ell(x)$ failures (3.3-2). The assumption of relative rates suggests that the $P^*(t|\underline{\ell})$ should approximate $S(t|\underline{\ell})$ and errs on the conservative side, overestimating failure probabilities. In the forward integral equations for $Q(t|\underline{\ell})$, $P(t|\underline{\ell})$ is replaced by $P^*(t|\underline{\ell})$ to yield an approximate solution for $Q(t|\underline{\ell})$ (equation 3.3-4). This again errs on the conservative side, overestimating $Q(t|\underline{\ell})$. Thus the reliability R(t) is underestimated. The error in these approximations for extremely small probabilities is under current investigation.

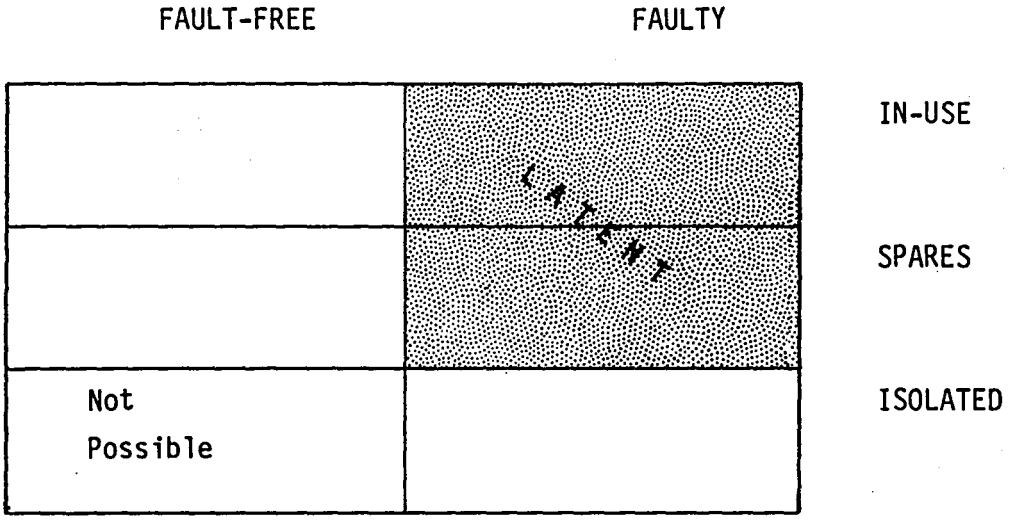
2.5 COVERAGE MODEL

Each module may be subject to several failure modes, each with a different rate of occurrence. The coverage model represents probabilistically the fault tolerance part of the system. Different types of faults may have different probabilities of occurrence and different probabilistic coverage models.

Modules are classified as either fault-free or faulty (Figure 2.5-1). Modules are also classified as being either in-use, spare or isolated. A module has a latent failure if the module is faulty and in-use or spare. The coverage model addresses the ability of the system to survive latent in-use failures, the object of the fault tolerance of the system.

There are nine states in the coverage model for a single module, as shown in Figure 2.5-2. The states are:

A	(active)	- module capable of producing an error
A_E	(active error)	- module producing error(s)
B	(benign)	- module has not produced errors and is currently not faulty
B_E	(benign error)	- module has produced error(s) but is currently not faulty
F	(failed)	- module has caused system failure
A_D	(active detected)	- module in active state has been detected as faulty
B_D	(benign detected)	- module in benign state has been detected as faulty
DP	(detected as permanent)	- module has been detected as having a permanent or intermittent fault and has been isolated from the system.



Module Classes

Fault-free : have not experienced a fault

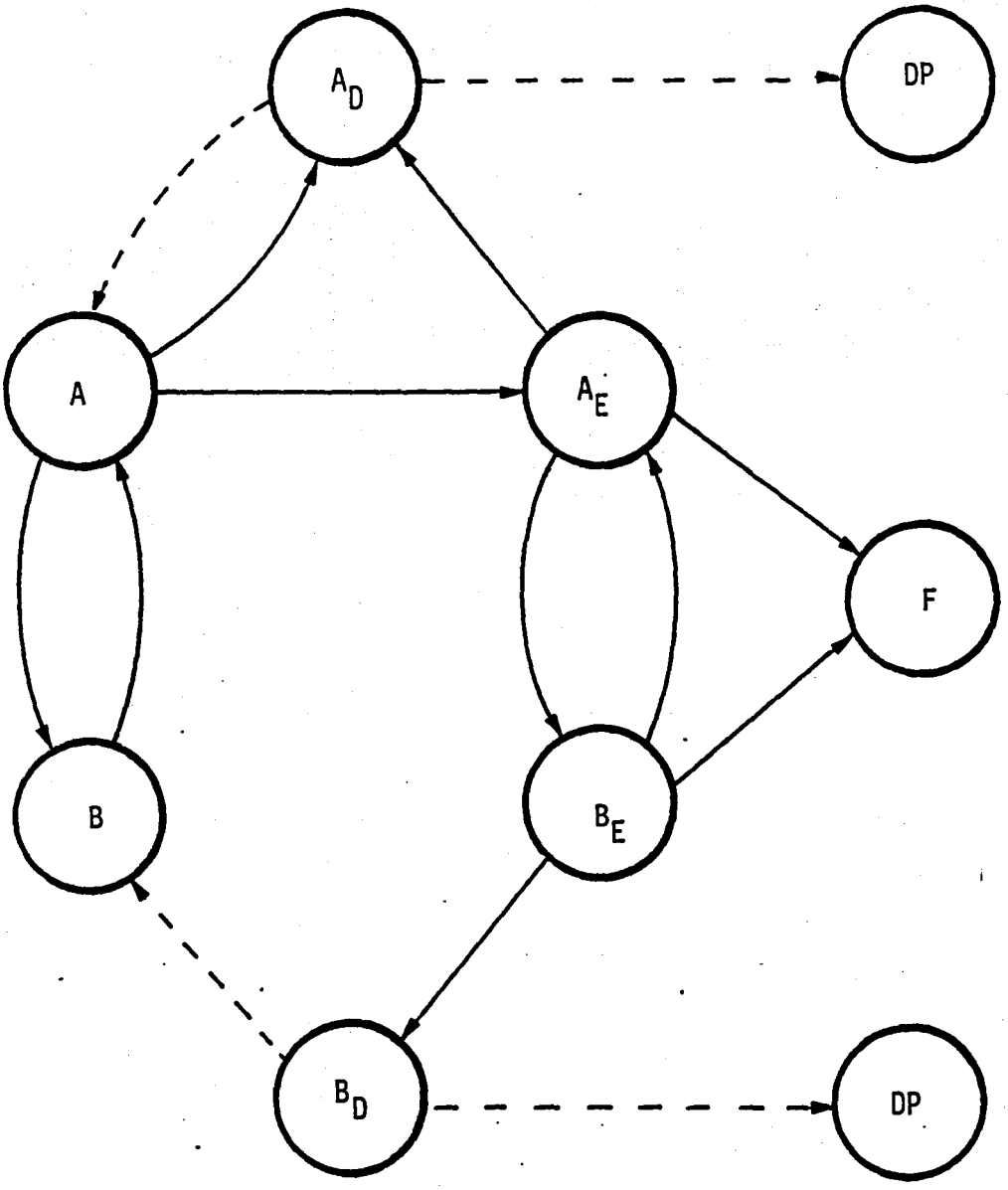
Faulty : have experienced a fault

In-use : active part of system

Spare : ready for, but not currently a part of active system

Isolated : deleted permanently from the active system.

Figure 2.5-1 Module Classification



A: ACTIVE
B: BENIGN
D: DETECTED
E: ERROR
F: FAILURE
DP: DETECTED AS PERMANENT (NON-TRANSIENT)

Figure 2.5-2 Single Fault Coverage Model

Three basic fault types are represented by the coverage model: permanent, intermittent and transient faults. A permanent fault, Figure 2.5-3, remains faulty and is either eventually detected and isolated (DP) or causes system failure (F), the only absorbing states. An intermittent fault, as shown by Figure 2.5-2, oscillates between the active and benign states (A and B or A_E and B_E) before it is isolated (DP) or causes system failure (F). A transient fault, Figure 2.5-4, is a temporary fault for a module. The module may cause system failure (F), be incorrectly identified as a permanent failure and isolated (DP), or become error free (B).

Two types of coverage failures are represented in CARE-III. A single fault coverage failure occurs when a faulty module propagates an error before the module is detected and isolated. A double fault coverage failure occurs when a pair of in-use modules with latent faults exist which together cause system failure. The double fault coverage model may be looked at as a combination of two single fault coverage models. Of the 9X9 = 81 potential states, various combinations of single fault non-failure states are double fault system failure states (AA, AB_E and A_EB). Some combinations are not possible, while others revert to the single fault coverage model.

When specifying fault modes for a stage, the specification of the transition rates between the coverage states defines the fault type. The critically coupled modules, within or between stages, which may cause a double fault failure are user specified by a critical pairs fault-tree. The single and double fault models are discussed in Sections 3.1.3 and 3.1.4. Derivation of the coverage rates for the reliability equations are given in Section 4.

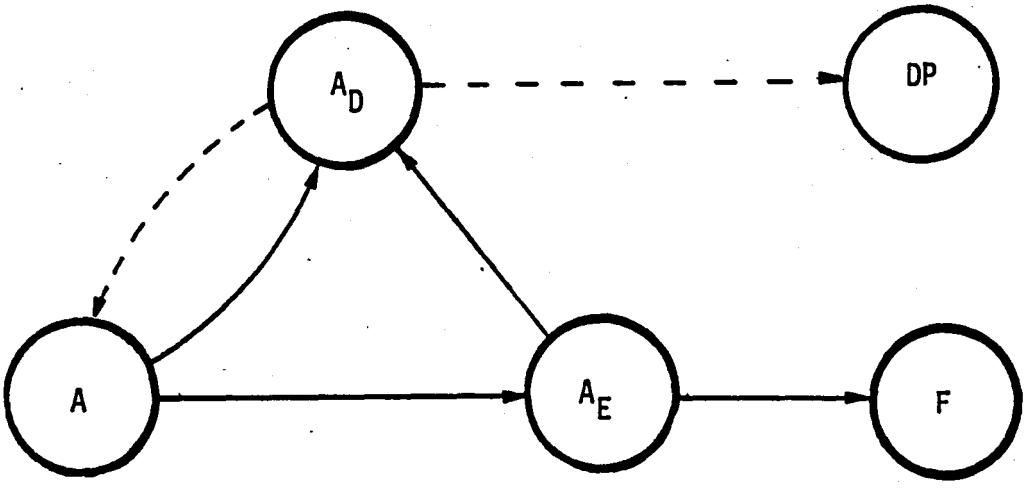
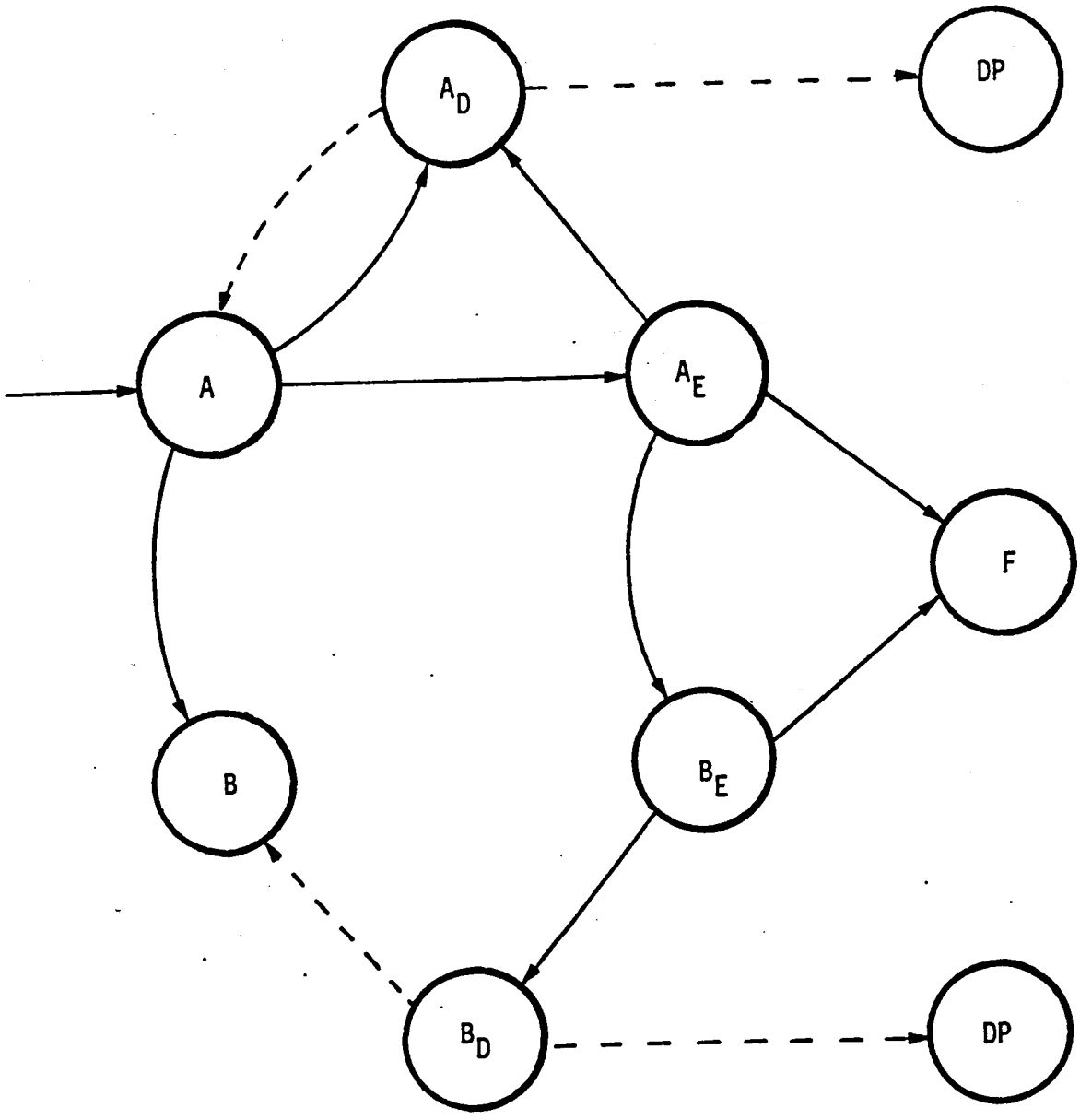


Figure 2.5-3 Single Permanent Fault Coverage Model



A: ACTIVE
B: BENIGN
D: DETECTED
E: ERROR
F: FAILURE
DP: DETECTED AS PERMANENT

Figure 2.5-4 Single Transient Fault Coverage Model

Section 3
RELIABILITY MODEL

The theory and implementation of the CARE-III reliability model is described in this section. The mathematical details of the reliability model have been extracted from the CARE-III documentation, J. J. Stiffler, L. A. Bryant and L. Guccione (1979), J. J. Stiffler, and L. A. Bryant (1982), J. J. Stiffler, J. S. Neumann and L. A. Bryant (1982) and the CARE-III program (Version-3, 1982). In those areas where the documentation is vague or incomplete, BCS has completed the model specifications based on its understanding of the applicable reliability methods.

Section 3.1 presents a specification of the detailed CARE-III reliability model including the concepts of spares exhaustion, and single or double fault failures; the model is characterized by a detailed state space model. The solution of the CARE-III model is described in Sections 3.2 to 3.4; first the aggregation procedure used to reduce the order of the reliability model is presented in Section 3.2, then the solution procedure used to solve the model is given in Section 3.3, and finally, the calculation of the transition rates for the model is described in Section 3.4. The implementation of the CARE-III model is described in Sections 3.5 to 3.6; first an overview of the CARE3 program is presented in Section 3.5, and then the computational methods used in CARE3 are highlighted in Section 3.6.

Any discrepencies between the CARE-III documentation or the CARE3 program and the reliability model, as found by BCS during the Task 1 review, are pointed out in the discussion in Section 3.

3.1 DETAILED RELIABILITY MODEL

3.1.1 Model Specifications

The objective of the model is to calculate the reliability of a complex, redundant Fault Tolerant System. Such highly reliable systems can fail due to exhaustion of adequate resources, but the dominant cause of failure tends to be failure to detect and isolate malfunctioning elements - coverage failures.

The system modeled by CARE III consists of a number of stages (up to 70), and each of these is composed of one or more identical interchangeable modules.

Recall the notation introduced in Section 2.2 where for stage-x,

(x,a) = a-th module in stage-x,
$n(x)$ = number of modules in stage-x,
$m(x)$ = minimum number of modules necessary for stage-x to be operational.

Furthermore, for each stage-x, the vector NOP(x) indicates the number of modules in-use as a function of the number of latent and fault free modules. NOP(x) is a vector of integers $(q(1,x), q(2,x),...)$ with $n(x) \geq q(i,x) > q(i+1,x) \geq m(x)$. If s modules have been deleted, and $q(i-1,x) > n(x) - s \geq q(i,x)$ $(q(o,x) = n(x)+1)$, then $q(i,x)$ modules are in-use, and $n(x) - s - q(i,x)$ modules are treated as operational spares.

Figure 2.5-1 shows the relationships between faulty, latent, in-use and spare modules.

In the present model each stage-x module can suffer from one of several categories of faults (up to 5). The j-th fault category for a stage-x module is denoted by x_j and is defined by its rate of occurrence,

$$\lambda(t \mid x_j) = \lambda(x_j)\ \omega(x_j) t^{\omega(x_j)-1}$$

and the fault coverage parameters (which characterizes whether the fault is permanent, intermittent or transient, its detection and isolation schedules, etc.).

A permanent or intermittent fault is said to be latent from the time it first occurs until it is detected and isolated from the system. A transient fault is said to be latent from the time it first occurs until it is either detected and isolated or reaches a benign state.

As discussed in Section 2.2 there are several causes of coverage failure:

Single Fault Failure

C1. An existing latent fault causes the system to take some unacceptable action,

Double Fault Failures

C2. A new fault occurs which, in combination with an existing latent fault, prevents the system from functioning properly;

C3. A pair of existing latent faults for the first time reaches a system-disabling state.

The analysis of the first cause of coverage failure and of the latency period of a fault is based on the Single Fault Coverage Model described in Section 3.1.3.

The last two causes of failure are analyzed in Section 3.1.4 and are applicable only for the case of interacting modules, i.e., critical pairs of modules. The set of such pairs denoted by CP depends on the architecture of the system and is defined by a Critical Pairs Fault Tree.

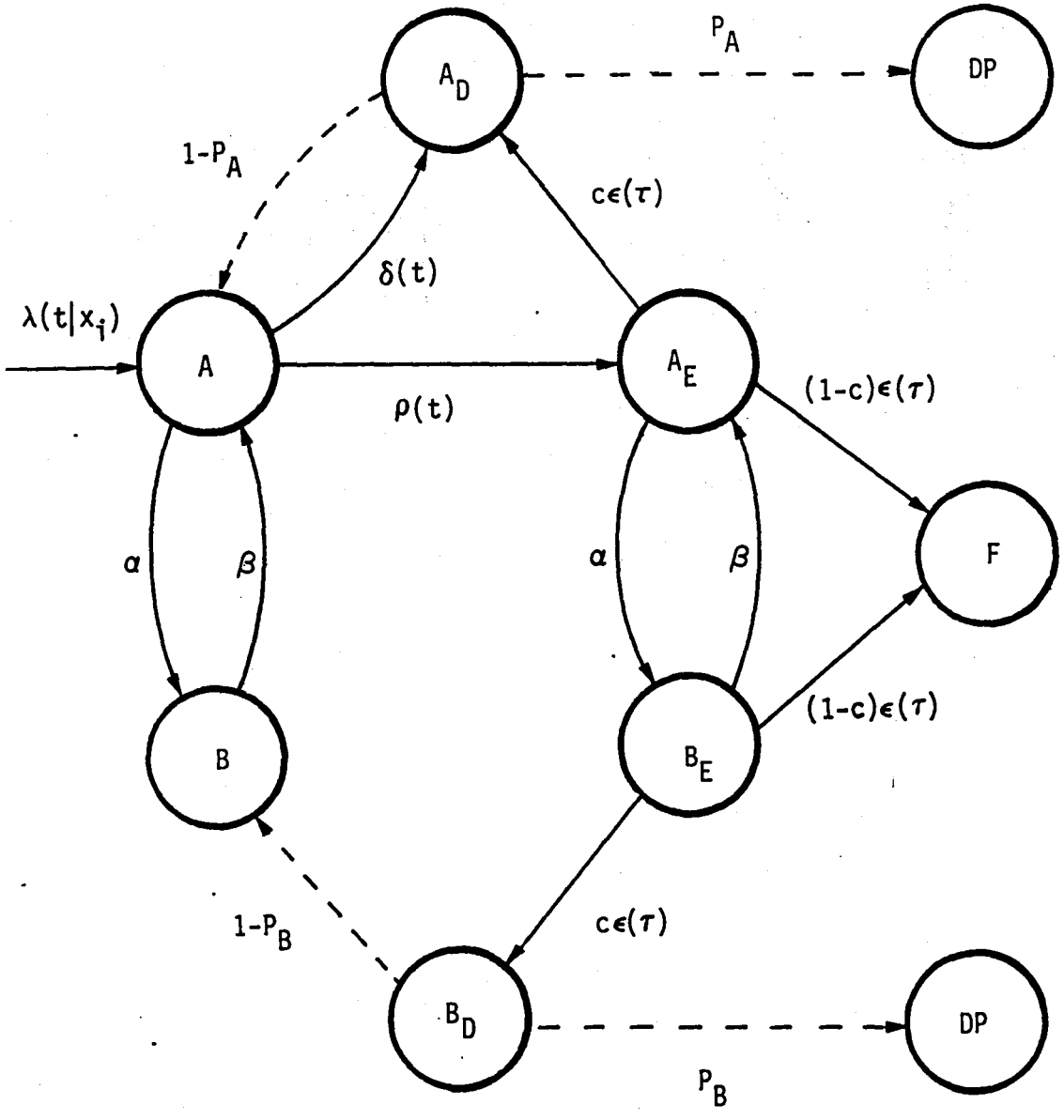
3.1.2 Spares Exhaustion Failure

The state of the system is represented by the vector $\underline{\ell} = (\ell(1), \ell(2),..., \ell(x),...)$, where $\ell(x)$ indicates the number of stage-x modules that have experienced a fault. Each stage-x is said to have failed (due to spares exhaustion) if the number of operational modules falls below the allowed minimum, i.e., $n(x) - \ell(x)$ is less than $m(x)$. System failure due to exhaustion of spares is then defined by a combination of stage failures introduced by a System Fault Tree. The set of spares exhaustion states is denoted by $\overline{L}$.

3.1.3 Single Fault Coverage Failure

The Single Fault Coverage Model, SFCM, defines the coverage structure and helps analyze the latency period of a fault. The SFCM is shown in Figure 3.1-1, and its dynamics are described in what follows.

When a fault first occurs, it is said to be in the active state A (see Figure 3.1-1). If the fault is transient or intermittent, it may jump from the active to the benign state B. These transitions take place at a constant rate α; for permanent faults, $\alpha = 0$. If the fault is intermittent, the reverse, benign-to-active, transition takes place at some constant rate β ; for transient faults, $\alpha \neq 0$ and $\beta = 0$. In the benign state, the fault is incapable of causing any discernable malfunction. Thus, it can neither be detected nor can it produce erroneous output. In the active state, however, the fault is both detectable and capable of producing incorrect output. The rates at which either of these events take place depend upon the operating environment and, in particular, on how frequently and how often the faulty element is exercised in a way that



A: ACTIVE
B: BENIGN
D: DETECTED
E: ERROR
F: FAILURE
DP: DETECTED AS PERMANENT (NON-TRANSIENT)

Figure 3.1-1 Single Fault Coverage Model

causes the defect to manifest itself. If the fault is detected the system enters the active-detected state A_D, and if it produces an error it enters the active-error state A_E. These transitions occur at time t, as measured from last entry into state A, according to the probability density functions $\delta(t)$ and $\rho(t)$ respectively. Once the system is in the active-error state A_E, if the fault is either intermittent or transient, it may jump to the benign state. The error is still present so the state is designated the benign-error state B_E. The composition of the two error states, A_E and B_E, is denoted the error state E. When the faulty element is in the error state, it jumps t time units after entry into A_E, and according to probability density function $\epsilon(t)$, to some point in the system. At this point the error is either detected or else propagates resulting in system failure, i.e., enters state F (coverage failure C1). The probabilities of these two alternatives are C and 1-C, respectively. If the fault is detected, either through testing or through the detection of an erroneous output, the faulty element enters the active-detected state A_D or benign-detected state B_D, depending on the state of the fault when it was detected. At that time a decision is made as to whether the faulty element is to be retired from the system or whether it can continue to be used. This latter decision might be made, for example, if the fault recovery procedure included a diagnostic routine designed to distinguish between permanent and transient faults. If the fault is detected in the active state, the decision is made with probability P_A that the element must be retired from service; if it is detected in the benign state, the same decision is made with probability P_B. In either case the process jumps to detected as permanent state DP. Thus, with probabilities $1-P_A$ and $1-P_B$, respectively, the faulty element is returned to service following the detection of the fault. (The dashed lines in Figure 3.1-1 indicate that the transition takes place immediately with the probability indicated.)

The model assumes that the effect of a decision that the fault is transient is to eliminate the error, if an error had already been produced, and to return the faulty element to the error-free, active or benign state, depending on its state when the fault was detected. If the fault

was transient and detected in the benign state, it returns to the error-free benign state. Since $\beta = 0$, it can never again become active so it ceases to pose any further threat to the system. If the fault is transient and detected as transient in the active state, it remains latent and may have another chance to cause system failure. If the fault is permanent or intermittent and detected as transient in either detected state, A_D or B_D, it remains latent and may have another chance to cause system failure.

In summary, in the SFCM,

- the states are:

A	active
B	benign
A_D	active detected
B_D	benign detected
A_E	active error
B_E	benign error
E	error (combination of A_E and B_E)
DP	detected as permanent
F	failure

- the parameters for the transitions are:

α	transition rate from A to B or from A_E to B_E
β	transition rate from B to A or from B_E to A_E
δ (t)	p.d.f. for transition from A to A_D, where t is measured from time of last entry into A
ρ (t)	p.d.f. for transition from A to A_E, where t is measured from time of last entry into A
ϵ (t)	p.d.f. for transition out of E, where t is measured from time of last entry into E
C	probability that a propagated error is detected before it causes the system to malfunction

P_A probability that a fault detected in the active state is detected as permanent

P_B probability that a fault detected in the benign state is detected as permanent

In the present version of CARE III, the functions $\delta(t)$, $\rho(t)$ and $\epsilon(t)$ are restricted to be either exponential or uniform densities, i.e., either of the form

$$\theta \exp(-\theta t), \quad t > 0,$$

or

$$\theta, \quad 0 < t < 1/\theta,$$

for some constant θ. However, the results obtained in Sections 3.2, 3.3, 3.4 and 4.2 are valid even when these functions are arbitrary densities with support on the positive time axis.

The transitions in this process occur either at constant rates, instantaneously or according to some density functions. The parameters that govern these transitions are independent of the time at which they occur: time homogeneous process. The transitions governed by the densities $\delta(t)$, $\rho(t)$ and $\epsilon(t)$, are assumed to be independent of past dynamics given that time t is measured from the time of last entry into states A or E: Markov property at jump times. It follows then that the Single-Fault Coverage Model satisfies the conditions of a Semi-Markov process (see Appendix A). If all three densities $\delta(t)$, $\rho(t)$ and $\epsilon(t)$ are exponential, then the SFCM is a Markov process. In Section 4.2 properties of such processes are used to calculate the state probabilities and intensities of entry that are needed to solve the reliability problem.

3.1.4 Double Fault Coverage Failure

The dynamics that lead to failures due to interacting modules can be based on the corresponding pair of Single Fault Coverage Models, and by determining if, and when, the two independent fault states form some lethal

combination. CARE III takes a simplified conservative approach described in the following paragraph and summarized in Table 3.1-1.

When the second module experiences a fault, the first module can be in any coverage state. If the first module is either active A or in error E the system is assumed to fail immediately (coverage failure C2); if the first module created errors that escaped undetected (F) the system has already failed and all future analysis is irrelevant; if the first module has been deleted from the system (DP) future dynamics of the system are independent from it. If the first module is benign B, the system enters the Double-Fault Coverage Model as described in the following paragraphs.

Double Fault Coverage Model

When the first module is in the benign B state the analysis is based on the Double Fault Coverage Model, DFCM, which corresponds to a simplified version of a combination of the corresponding single models. The DFCM is shown in Figure 3.1-2.

The DFCM is entered according to a rate that depends on the first fault being benign and on the rate of occurrence of the second fault. Such a situation places the fault-pair in the B_1A_2 state (first fault benign, second fault active.). From there, the fault-pair can go to the B_1B_2 state (both faults benign) if the second fault becomes benign before the first fault becomes active, to the state B_1D_2 if the active fault is detected or to the failed state DF (double fault) if the first fault becomes active with the second fault still also in the active state or if the second fault causes an error to be produced (coverage failure C3). From the state B_1D_2 it can either go to B_1DP_2 or return to B_1A_2 depending on whether the second is detected as permanent or not. In the state B_1DP_2 the second module is deleted from the system. If the first module is interacting with some other latent module, it continues to be analyzed within the context of the DFCM. If not, it is analyzed in the context of the SFCM.

TABLE 3.1-1
COVERAGE FAILURES FOR CRITICAL PAIRS

Coverage state of first module when second fault becomes active	Consequence	Future analysis	Type of coverage Failure
A, A_E or B_E	Immediate failure	---	C2
F	System already failed	---	C1
DP	First module deleted from system	Independent of first module	---
B	Interaction of faults	Enters DFCM	Possible C3

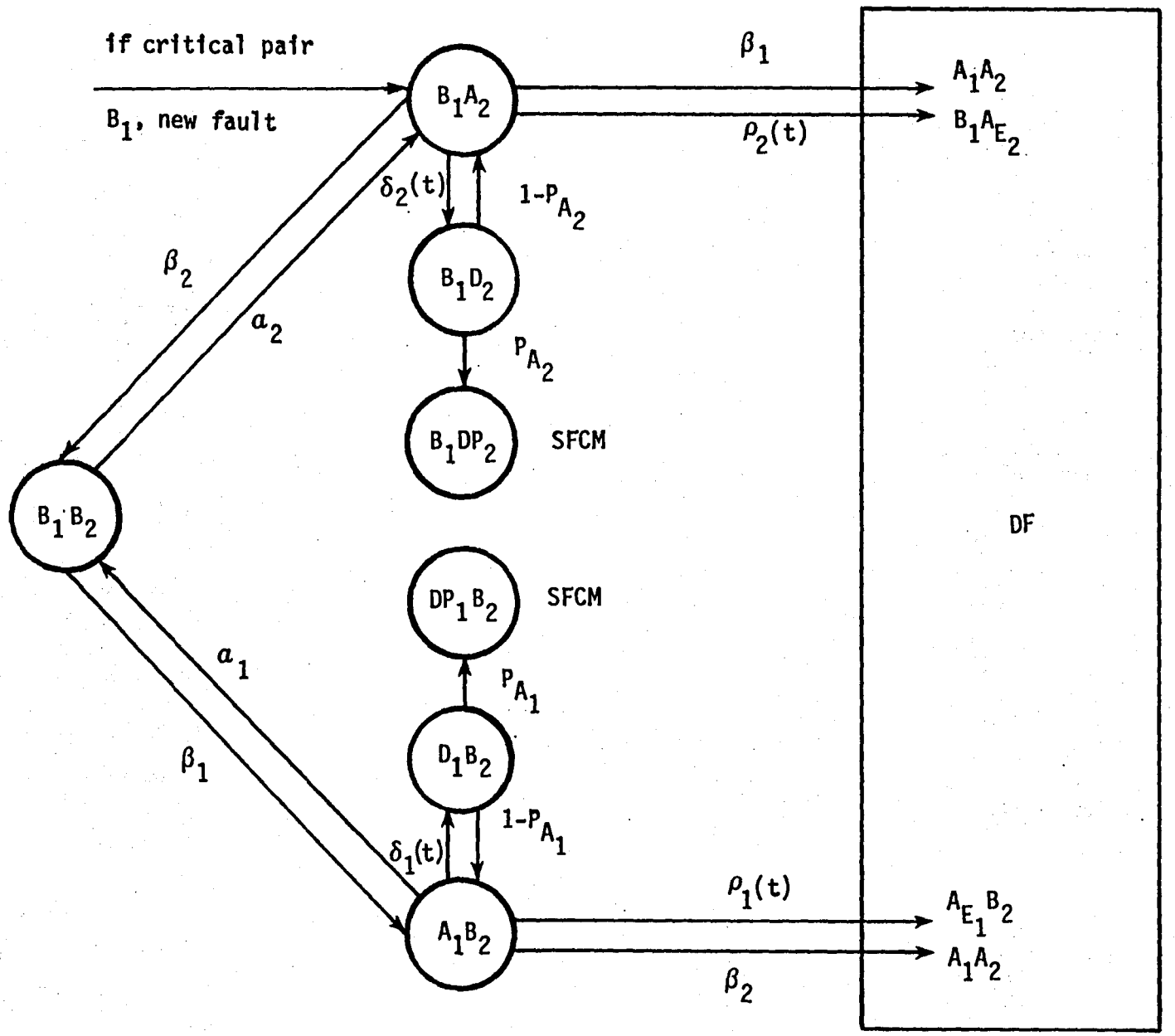


Figure 3.1-2 Double Fault Coverage Model

Since both faults are benign in the B_1B_2 state, the only possible transitions from that state are back to the A_2B_1 state or to the A_1B_2 state (first fault active, second fault benign) with its entirely analogous transitions.

The argument used for the SFCM shows that the Double Fault Coverage Model is a Semi-Markov process. The intensity entry into state DF characterizes coverage failures of the third type, C3, and its formula is derived in Section 4.2.

3.1.5 State Space Definition

The state of the system is determined at each time by the status of each module:

- A fault has occurred or not;
- Category that caused the fault (if fault occurred);
- Coverage fault status (if fault occurred): active, benign, detected, etc;
- Spare status of a latent or fault free module.

Analytically the states are described by the three M-dimensional vectors $\underline{d},\underline{i}$ and $\underline{c}$, which have been defined in Section 2.2. The information given by these vectors corresponds respectively to occurrence of faults, fault categories and coverage status.

Part of the information given by the triple $(\underline{d},\underline{i},\underline{c})$ is summarized in the vectors $\underline{\ell} = (\ell(1), \ell(2),\ldots, \ell(x),\ldots)$ and $\underline{\mu} = (\mu(1), \mu(2),\ldots, \mu(x), \ldots)$ where $\ell(x)$ and $\mu(x)$ denote the numbers of stage-x faulty and latent modules, respectively. The triple $(\underline{d},\underline{i},\underline{c})$ gives no information on which latent and fault free modules are in-use and which are spare. This information shall be assumed implicit, and will be used in the classification of states, description of transitions and evaluation of aggregate rates.

Classification of States

The states of the model can be classified according to the failure status of the system:

- The set of spares exhaustion states is denoted by $\overline{L}$ and is defined as some combination of unions and/or intersections of sets of the form

$$\left\{(\underline{d},\underline{i},\underline{c}) \;/\; n(x) - \ell(x) < m(x)\right\} .$$

  This last set corresponds to all states for which stage-x has fewer than m(x) operational modules (i.e., "failure" of stage-x). Such a $(\underline{d},\underline{i},\underline{c})$ state shall be said to be an H state.

- L denotes the complement of $\overline{L}$.

A state $(\underline{d},\underline{i},\underline{c})$ in L, although not defining failure due to spares exhaustion, can represent a case of coverage failure. To isolate such cases consider the latent in-use modules determined by $(\underline{d},\underline{i},\underline{c})$, i.e., non-deleted non-spare faulty modules. Of these consider all possible latent critical pairs and all other latent modules (called single modules hereafter).

The state $(\underline{d},\underline{i},\underline{c})$ is an F (failure) state if either there is a latent single module that created an undetected error (i.e., c-component is F), or there is a latent critical pair in state DF, as given in Figure 3.1-2.

If none of the above conditions are satisfied, the state $(\underline{d},\underline{i},\underline{c})$ is called a G (operational) state.

3.1.6 Stochastic Characteristics of the Model

The stochastic model for this state space is characterized by a mixture of

- Time dependent rates for occurrence of faults;
- Semi-Markov processes on coverage dynamics.

Backward integral equations, similar to those given in the Appendix, could be used to calculate the probabilities that the system is in an operational state. The reliability of the system would then obtained by adding all these.

This approach, though straightforward, has the disadvantage of requiring such a large number of calculations that even for moderate size systems the problem becomes unmanageable.

Two steps are taken to obtain a feasible solution. First, states with the same number of faults and similar failure characteristics are collected into aggregate states, thus reducing the size of the state space. Second, detailed information given by the individual states forming an aggregate state is replaced by probabilistic statements, thus allowing a decomposition between the Coverage Models and the Aggregate Reliability Model. The Coverage Models are used to derive the transition rates for the Aggregate Reliability Model, which, under certain conditions on holding times of the original process, is solved as a non-homogeneous Markov process.

In Section 3.2 the Aggregate Model will be described in detail, excluding only the case of transient faults. As mentioned in J. J. Stiffler, J. S. Neumann and L. A. Bryant (1982), these cases showed instabilities. A model with transient faults would include backward transitions, e.g., from $\underline{\ell}$ to $\underline{\ell} - \underline{1}(x)$ if some transient stage-x fault becomes benign. This step is avoided in the present version of CARE III by an approximation based on the relative speeds of coverage rates and occurrence of fault

rates. As an example consider a transient fault as given in Figure 3.1-3a, i.e., the fault is either detected or becomes benign with constant rates δ and a, respectively. The probability of being in state D within the coverage model is given by

$$P_D(t) = \frac{\delta}{a + \delta} \quad 1 - \exp(-(a+\delta)t) \quad .$$

Figure 3.1-3b shows the dynamics of this fault including the occurrence of the fault and its possible recurrences. The probability of being in state D within the whole system is then given by

$$H_D(t) = \int_0^t \frac{\lambda \delta}{s_2 - s_1} \exp(-s_1 \mu) - \exp(-s_2 \mu) \quad du,$$

where s_1 and s_2 are the roots of

$$s^2 - (\lambda + a + \delta)\, s + \lambda\delta = 0.$$

If the parameter δ is much larger than λ, then $H_D(t)$ can be approximated by

$$\int_0^t \lambda P_D(t - u)du.$$

This last integral corresponds to the value of $H_D(t)$ as presently calculated in CARE III.

The effect of this approach to handling transient faults on the calculation of transition rates and state probabilities in the Aggregate Model is not fully understood. This problem and the instability encountered by Raytheon in Phase III is not addressed in this report.

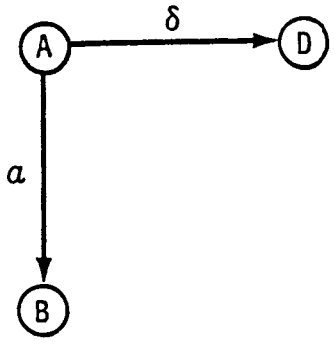


Figure 3.1-3a Example of Transient Fault

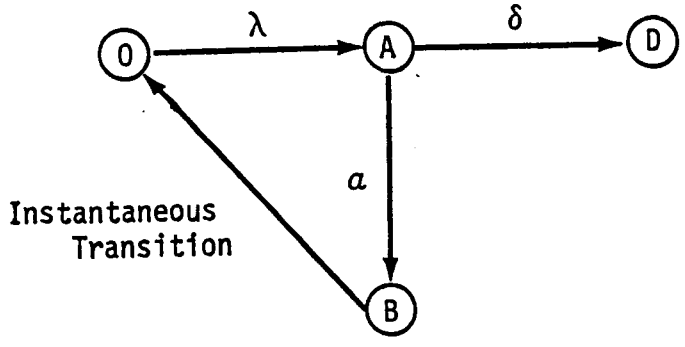


Figure 3.1-3b Dynamics of Transient Fault

3.2 REDUCED RELIABILITY MODEL

3.2.1 Aggregate States

The reduced state space is formed by aggregation of states with identical number of faults per stage and similar failure characteristics.

As defined in Section 2.3 the possible aggregates for each fault vector are:

If $\underline{\ell}$ is in $\overline{L}$,
$H(\underline{\ell})$: aggregate of spares exhaustion states with $\underline{\ell}$ faults;

and if $\underline{\ell}$ is in L,
$G(\underline{\ell})$: aggregate of operational states with $\underline{\ell}$ faults,
$F(\underline{\ell})$: aggregate of failure states with $\underline{\ell}$ faults.

3.2.2 Transitions and Rates

As discussed in Section 2.3 the transitions and rates in the Aggregate Model are:

- From $H(\underline{\ell})$ to $H(\underline{\ell} + \underline{1}(y))$ with rate $\lambda^{*}(t|\underline{\ell}, \underline{\ell}+\underline{1}(y))$, if a fault occurs on a fault free stage-y module;

- From $F(\underline{\ell})$ no transitions are possible since these states are absorbing;

- From $G(\underline{\ell})$ to $F(\underline{\ell})$ with rate $\mu(t|\underline{\ell})$ if a coverage failure occurs;

- From $G(\underline{\ell})$ to either
  $H(\underline{\ell}+\underline{1}(y))$ with rate $\lambda^{*}(t|\underline{\ell}, \underline{\ell}+\underline{1}(y))$,
  $F(\underline{\ell}+\underline{1}(y))$ with rate $\lambda^{(2)}(t|\underline{\ell}, \underline{\ell}+\underline{1}(y))$, or

$G(\underline{\ell}+\underline{1}(y))$ with rate $\lambda^{(1)}(t|\underline{\ell},\underline{\ell}+\underline{1}(y))$,
if a fault occurs on a fault free state-y module.

To determine under which conditions the different transitions out of $G(\underline{\ell})$ occur, it is necessary to disaggregate this state and to analyze the dynamics within each of its parts.

Fix a state $(\underline{d},\underline{i},\underline{c})$ in $G(\underline{\ell})$. Such a state divides latent in-use modules into two groups:

- Interacting modules: those latent in-use modules which form a critical pair with another latent in-use module. Such a pair of modules will be called an interacting pair.

- Single modules: those latent in-use modules which are not interacting.

The possible transitions out of the state $G(\underline{\ell})$ are then as follows:

(1) To $F(\underline{\ell})$ if either

(1.a) A single module created an error that escaped undetected (i.e., transition from E to F in corresponding SFCM), or

(1.b) An interacting pair, with one active module and the other benign, and either the active module created an error or the benign module became active (i.e., transition from AB to DF in corresponding DFCM).

(2) To $H(\underline{\ell}+\underline{1}(y))$ if both

(2.a) A fault occurs on a fault free stage-y module, and

(2.b) $\underline{\ell}+\underline{1}(y)$ is in $\bar{L}$.

(3) To F($\underline{\ell}+\underline{1}(y)$) if the following four conditions hold:

(3.a) A fault occurs on a fault free stage-y module,

(3.b) $\underline{\ell}+\underline{1}(y)$ is in L,

(3.c) The new faulty module is in-use, and

(3.d) A latent in-use module, critically paired with the new faulty module, and either

(3.d.1) Is single and non-benign (i.e., active or in error), or

(3.d.2) Is interacting and active.

(4) To G($\underline{\ell}+\underline{1}(y)$) if the following three conditions hold:

(4.a) A fault occurs on a fault free stage-y module,

(4.b) $\underline{\ell}+\underline{1}(y)$ is in L, and

(4.c) Either

(4.c.1) The new faulty module is not in-use (i.e., it is a spare module), or

(4.c.2) No latent in-use module satisfies condition (3.d).

An analysis of the above conditions leads to the following observations:

- Given that (3.a) holds, condition (4.c) is complementary to the set of conditions (3.c) and (3.d), and so

$$\lambda^{(1)}(t|\underline{\ell}, \underline{\ell}+\underline{1}(y)) + \lambda^{(2)}(t|\underline{\ell}, \underline{\ell}+\underline{1}(y)) = \lambda^{*}(t|\underline{\ell}, \underline{\ell}+\underline{1}(y));$$

- The transitions from $G(\underline{\ell})$ to $H(\underline{\ell}+\underline{1}(y))$, from $H(\underline{\ell})$ to $H(\underline{\ell}+\underline{1}(y))$, and from $\underline{\ell}$ to $\underline{\ell}+\underline{1}(y)$ under perfect coverage, depend on the vector $\underline{\ell}$ of faults but not on the coverage status of the system. Thus the rate for all three transitions is given by

$$\lambda^{*}(t|\underline{\ell}, \underline{\ell}+\underline{1}(y)) = (n(y)-\ell(y)) \sum_{j} \lambda(t \mid y_j)$$

In the present version of CARE-III, the conditions that define transitions from $G(\underline{\ell})$ to $F(\underline{\ell}+\underline{1}(y))$ are replaced by

(3.a') A fault occurs on a fault free stage-y module,

(3.b') $\underline{\ell}+\underline{1}(y)$ is in L,

(3.c') The new faulty module is in-use, and

(3.d') A latent in-use and non-benign module is critically paired with the new faulty module.

These new conditions allow for a simpler evaluation of the rates $\lambda^{(2)}$, and lead to a conservative value of the reliability of the system.

3.2.3 Assumptions on Stochastic Properties

The original stochastic model is a mixture of time dependent rates for occurrence of faults and Semi-Markov process for coverage dynamics.

The discussion in Section 2.2 on the comparative speed of coverage dynamics with that of occurrence of faults suggests that internal transitions within aggregate states occur at much faster rates than those between such states. Thus the dynamics within these states can be assumed to happen instantaneously and the Aggregate Model is well approximated by a non-homogeneous Markov process.

3.2.4 Model Equations and Solutions

Assuming that the rates for the Aggregate Model are known (their derivation is given in Section 3.4), the results given in the Appendix for non-homogeneous Markov processes are used to calculate state probabilities and the reliability of the system.

Let the state probabilities be:

$P(t|\underline{\ell})$: probability that the system is in state $G(\underline{\ell})$ at time t;

$Q(t|\underline{\ell})$: probability that the system is in state $F(\underline{\ell})$ at time t;

$S(t|\underline{\ell})$: probability that the system is in state $H(\underline{\ell})$ at time t.

These probabilities are conditional on the system being fault free at time 0.

The corresponding forward differential equations are then:

$$\frac{d}{dt} P(t|\underline{\ell}) = -P(t|\underline{\ell})\,\lambda(t|\underline{\ell}) + \sum_x P(t|\underline{\ell}-\underline{1}(x))\,\lambda^{(1)}(t|\underline{\ell}-\underline{1}(x),\underline{\ell}), \tag{3.2-1}$$

$$\frac{d}{dt} Q(t|\underline{\ell}) = P(t|\underline{\ell})\,\mu(t|\underline{\ell}) + \sum_x P(t|\underline{\ell}-\underline{1}(x))\,\lambda^{(2)}(t|\underline{\ell}-\underline{1}(x),\underline{\ell}), \tag{3.2-2}$$

with

$$\lambda(t|\underline{\ell}) = \mu(t|\underline{\ell}) + \sum_x \overset{*}{\lambda}\left(t|\underline{\ell}, \underline{\ell}+\underline{1}(x)\right),$$

$$\frac{d}{dt} S(t|\underline{\ell}) = -S(t|\underline{\ell})\ \overset{*}{\lambda}(t|\underline{\ell}) + \tag{3.2-3}$$

$$+ \sum_x \left[P(t|\underline{\ell}-\underline{1}(x)) + S(t|\underline{\ell}-\underline{1}(x))\right] \overset{*}{\lambda}(t|\underline{\ell}-\underline{1}(x), \underline{\ell}),$$

with

$$\overset{*}{\lambda}(t|\underline{\ell}) = \lambda(t|\underline{\ell}) - \mu(t|\underline{\ell}).$$

The equivalent forward integral equations are:

$$P(t|\underline{\ell}) = \sum_x \int_0^t P(u|\underline{\ell}-\underline{1}(x))\ \lambda^{(1)}(u|\underline{\ell}-\underline{1}(x), \underline{\ell})\ \exp\left[-\Lambda(u,t|\underline{\ell})\right] du, \tag{3.2-4}$$

$$Q(t|\underline{\ell}) = \int_0^t \left[P(u|\underline{\ell})\,\mu(u|\underline{\ell}) + \sum_x P(u|\underline{\ell}-\underline{1}(x))\ \lambda^{(2)}(u|\underline{\ell}-\underline{1}(x), \underline{\ell})\right] du, \tag{3.2-5}$$

and

$$S(t|\underline{\ell}) = \int_0^t \sum_x \left[P(u|\underline{\ell}-\underline{1}(x)) + S(u|\underline{\ell}-\underline{1}(x))\right] \overset{*}{\lambda}(u|\underline{\ell}-\underline{1}(x), \underline{\ell})\ \exp\left[-\overset{*}{\Lambda}(u,t|\underline{\ell})\right] du, \tag{3.2-6}$$

with Λ and $\overset{*}{\Lambda}$ denoting the definite integrals of $\lambda(.|\underline{\ell})$ and $\overset{*}{\lambda}(.|\underline{\ell})$ respectively.

The reliability of the system is then

$$R(t) = \sum_{\underline{\ell} \text{ in } L} P(t|\underline{\ell}), \tag{3.2-7}$$

or equivalently the unreliability of the system is

$$1 - R(t) = \sum_{\underline{\ell} \text{ in } L} Q(t|\underline{\ell}) + \sum_{\underline{\ell} \text{ in } \bar{L}} S(t|\underline{\ell}) \tag{3.2-8}$$

3.3 CARE III SOLUTION

3.3.1 Perfect Coverage Model

In Section 3.2 it was seen that the probabilities $P(t|\underline{\ell})$ needed to calculate the reliability of the system are obtained from a system of ordinary differential equations. These can be solved recursively by successive increments in the vector $\underline{\ell}$.

Since, as has been repeatedly observed in this discussion, the systems of concern here are highly reliable, $\lambda^{(1)}(t|\underline{\ell}, \underline{\ell}+\underline{1}(y))$ must in general be much larger than $\lambda^{(2)}(t|\underline{\ell}, \underline{\ell}+\underline{1}(y))$ and $\lambda(t|\underline{\ell})$ must be large compared to $\mu(t|\underline{\ell})$. Thus $\lambda^{(1)}(t|\underline{\ell}, \underline{\ell}+\underline{1}(y))$ is close to $\lambda^{*}(t|\underline{\ell}, \underline{\ell}+\underline{1}(y))$ and $\lambda(t|\underline{\ell})$ is close to $\lambda^{*}(t|\underline{\ell})$. So equation (3.2-1) can be replaced by

$$\frac{d}{dt} P^{*}(t|\underline{\ell}) = -P^{*}(t|\underline{\ell})\ \lambda^{*}(t|\underline{\ell}) + \sum_{x} P^{*}(t|\underline{\ell}-\underline{1}(x))\ \lambda^{*}(t|\underline{\ell}-\underline{1}(x), \underline{\ell}). \tag{3.3-1}$$

As mentioned in 3.2, the rates λ^{*} correspond to the perfect coverage case and so $P^{*}(t|\underline{\ell})$ given in (3.3-1) represents the probability of $\underline{\ell}$ faults at time t given perfect coverage.

Under perfect coverage, the interactions between stages are not relevant and so the fault status of stages are independent from each other. It follows that

$$P^{*}(t|\underline{\ell}) = \prod_{x} \binom{n(x)}{\ell(x)} \left[1-r(t|x)\right]^{\ell(x)} \left[r(t|x)\right]^{n(x)-\ell(x)} \tag{3.3-2}$$

where

$$r(t|x) = \exp\left[- \int_0^t \sum_i \lambda(u|x_i)\, du \right] = \qquad (3.3\text{-}3)$$

= reliability of a stage-x module.

Formula (3.3-2) can also be obtained directly by solving equation (3.3-1).

3.3.2 Approximate Reliability

The approach taken in CARE III is to calculate $Q(t|\underline{\ell})$ and $S(t|\underline{\ell})$ by using $P^*(t|\underline{\ell})$ instead of $P(t|\underline{\ell})$ in equations (3.2-2) and (3.2-3). The result of such approximation is to assume that the system has been operating under perfect coverage up to time t. This is represented in Figures 3.3-1a and b. Analytically it follows that $P^*(t|\underline{\ell})$ is larger than $P(t|\underline{\ell})$ and so the approximate values of $Q(t|\underline{\ell})$ and $S(t|\underline{\ell})$ are larger than those obtained from equations (3.2-2) and (3.2-3). Hence, a conservative value of the reliability is obtained.

The new equation for $S(t|\underline{\ell})$ can be shown to be equivalent to equation (3.3-1) and so $S(t|\underline{\ell})$ is approximated by $P^*(t|\underline{\ell})$.

The CARE III approach can be summarized by the following steps:

- calculate $P^*(t|\underline{\ell})$ using equation (3.3-2)
- calculate $Q(t|\underline{\ell})$ using

$$Q(t|\underline{\ell}) = \int_0^t \left[P^*(u|\underline{\ell})\, \mu(t|\underline{\ell}) + \sum_x P^*(u|\underline{\ell}\text{-}\underline{1}(x))\ \lambda^{(2)}(u|\underline{\ell}\text{-}\underline{1}(x), \underline{\ell}) \right] du \qquad (3.3\text{-}4)$$

- calculate the unreliability of the system by

$$1\text{-}R(t) = \sum_{\underline{\ell} \text{ in } L} Q(t|\underline{\ell}) + \sum_{\underline{\ell}' \text{ in } \bar{L}} P^{*}(t|\underline{\ell}) \qquad (3.3\text{-}5)$$

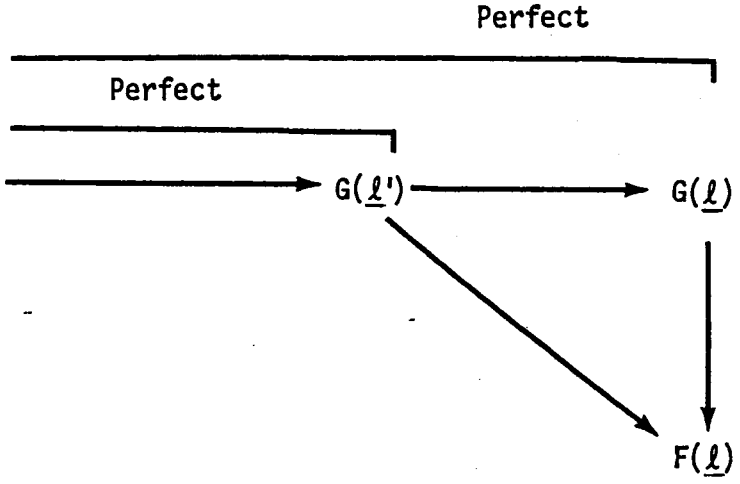


Figure 3.3-1a Approximations of State Probabilities $Q(t|\underline{\ell})$

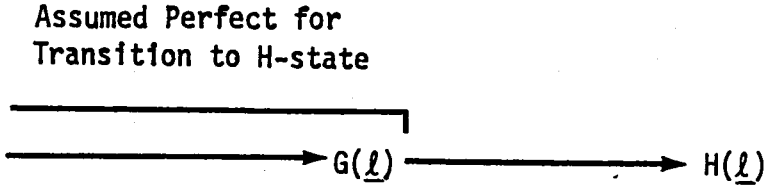


Figure 3.3-1b Approximations of State Probabilities $S(t|\underline{\ell})$

3.4 TRANSITION RATES

3.4.1 Approximations Used

In the original detailed description of the system it is known at each time which modules have experienced a fault, the category that caused the fault and the coverage status (active, benign, ...). Furthermore, it is known which of the faulty modules form critical pairs.

In the reduced model this level of detail has been lost. The rates of interest will then be obtained by first calculating the corresponding conditional rates given some detailed faulty structure and then integrating with respect to the probability distribution of such structure, i.e., $r(t) = E\left[r(t) \mid h_t\right]$, where $r(t)$ is the rate of interest and h_t is the detailed history of the process to time t.

In the rest of this section Y_t denotes the state of the system at time t. So $Y_t = A$ denotes the occurrence of event A at time t, and $P(Y_t = A)$ the corresponding probability.

Two basic properties are used to derive the aggregate rates:

P1 If A and B are any two states, where A is the aggregate of simpler states A_i, and if $r(t)$, $r_i(t)$ denote the rates corresponding to transitions from state A, A_i, respectively, to state B, then

$$r(t) \leq \sum_i r_i(t)\, P\left[Y_t = A_i \mid Y_t = A\right],$$

with equality if the states A_i are disjoint.

P2 Let $T_1, T_2, \ldots, T_n$ be independent and competing transition times that occur with rates $r_i(t)$. Then the transition rate $r(t)$ of the smallest transition time is given by the sum of the $r_i(t)$.

In the derivation of the formulas for the aggregate rates, property P1 can be used by disaggregating the $G(\underline{\ell})$ state into its $(\underline{d},\underline{i},\underline{c})$ components and considering all the possible choices of in-use (non-spare) modules. Those that are possible are not equally likely. Some choices would imply past failure of the system, e.g., if $(\underline{d},\underline{i},\underline{c})$ determines a critical pair of latent, in-use modules whose DFCM state is DF, this implies past system failure. To consider only the truly possible choices of in-use modules, given that the system is operational, and how each of these affects the aggregate rates, would entail an analysis of the detailed past history of the process. Such an approach defeats the purpose of the aggregation and decomposition steps taken to decrease the size of the state space.

The following assumption is thus made:

(A1) Given $\underline{\ell}$ faults in the system, all choice of spare, in-use groups of modules are possible, independently of the failure status of the system.

This assumption implies that all states $(\underline{d},\underline{i},\underline{c})$ with the same number of latent modules, and the same number of latent, in-use modules, are equally likely. Hence, all of these contribute in the same way to the aggregate rates. Since this procedure includes more critical pairs than are truly possible, over estimates of the aggregate rates are then obtained.

Condition A1 is based on the conservative assumption that the system has been operating under perfect coverage until the present. Furthermore, A1 is consistent both with the approach taken in Section 3.3 to calculate the failure state probabilities, $Q(t \mid \underline{\ell})$ and $S(t \mid \underline{\ell})$, and with the discussion given in J.J. Stiffler, L.A. Bryant and L. Guccione (1979, pp. 32-34).

3.4.2 Calculation of $\lambda^{(2)}(t)$

Let $\lambda^{(2)}(t \mid \underline{\ell}, \underline{\ell} + \underline{1}(y))$, or simply $\lambda^{(2)}(t)$, denote the rate of a transition at time t from state $G(\underline{\ell})$ to the failure state $F(\underline{\ell} + \underline{1}(y))$. Since $G(\underline{\ell})$ is the aggregate of states of the form $(\underline{d},\underline{c})$, using property P1 in Section 3.4.1, it follows that

$$\lambda^{(2)}(t) = \sum_{(d,c)} \lambda^{(2)}(t \mid \underline{d},\underline{c})\ P\left[Y_t = (\underline{d},\underline{c}) \mid Y_t = G(\underline{\ell})\right],$$

where $\lambda^{(2)}(t \mid \underline{d},\underline{c})$ denotes the rate of a transition from $(\underline{d},\underline{c})$ to $F(\underline{\ell} + \underline{1}(y))$. Such a transition occurs when the first fault free, in-use stage-y module, say (y,b), suffers the first possible fault category, say y_j; using property P2 it follows that

$$\lambda^{(2)}(t) = \sum_{(d,c)} \sum_{b,j} \lambda(t \mid y_j)\ P\left[\begin{array}{l} Y_t = (\underline{d},\underline{c}), \text{ and } (y,b) \\ \text{ is fault-free, in-use} \end{array} \;\middle|\; Y_t = G(\underline{\ell})\right].$$

As described in Section 3.2.2., a transition to failure state occurs if there is a non-benign, in-use module that is critically paired with the module (y,b).

Again using property P1, by choosing all possible in-use faulty modules (x,a) that form a lethal combination with (y,b) and all possible vectors $\underline{\mu}$ of latent modules, gives:

$$\lambda^{(2)}(t) \le \sum_{\mu} \sum_{(d,c)} \sum_{b,j} \sum_{x,a} \lambda(t \mid y_j)\ P\left[Y_t = (\underline{\mu},\underline{\ell}) \mid Y_t = G(\underline{\ell})\right]$$
$$P\left[Y_t = (\underline{d},\underline{c}), \text{ and } C(a,b) \;\middle|\; Y_t = (\underline{\mu},\underline{\ell})\right]$$

where C(a,b) denotes the occurrence of the event: module (x,a) is non-benign, module (y,b) is fault-free, and both form an in-use critical pair.

Assumption A1 implies that the status of stages are mutually independent, and depend on the number of faults $\underline{\ell}$, but not on the failure status of

the system. Furthermore, $\mu(x)$ and $n(y) - \ell(y)$ are upper bounds for the numbers of latent, in-use stage-x modules, and of fault-free, in-use, stage-y modules, respectively. It follows then that:

$$\lambda^{(2)}(t) \leq \sum_{\mu(x)\mu(y)} \sum_{j} \sum_{a} \lambda(t \mid y_j)\, \mu(x) \left[n(y) - \ell(y)\right] \frac{H_B^-(t \mid x)}{H_L(t \mid x)}$$

$$\cdot\, b^{(2)}_{x,y}(\underline{\mu}, \underline{\ell})\; P\left[Y_t = (\mu(x), \mu(y)) \mid Y_t = (\ell(x), \ell(y))\right]$$

where

$H_B^-(t \mid x)$: probability that a given stage-x module has a non-benign fault at time t;

$H_L(t \mid x)$: probability that a given stage-x module has a latent fault at time t;

$b^{(2)}_{x,y}(\underline{\mu}, \underline{\ell})$: probability that a given pair of (x,y) modules is critical when chosen from existing latent in-use stage-x modules, and fault-free in-use stage-y modules, given $\underline{\mu}$ latent and $\underline{\ell}$ faulty modules.

The H functions are evaluated by conditioning on the time of occurrence of the x_i fault. The $b^{(2)}_{x,y}$ function is evaluated by conditioning on the number of latent in-use modules in stages x and y.

In summary, the rate for a transition from $G(\underline{\ell})$ to $F(\underline{\ell} + \underline{1}(y))$ is given by

$$\lambda^{(2)}(t \mid \underline{\ell}, \underline{\ell} + \underline{1}(y)) = \left[\sum_{j} \lambda(t \mid y_j)\right] \left(n(y) - \ell(y)\right) c(t \mid y, \underline{\ell}), \quad (3.4\text{-}1)$$

with

$$c(t \mid y, \underline{\ell}) = \sum_{x} \frac{H_{\bar{B}}(t \mid x)}{H_{L}(t \mid x)} \; D(t,(x,y) \mid \underline{\ell}), \tag{3.4-2}$$

$D(t,(x,y)|\underline{\ell})$ is defined by (3.4-3)

$$\sum_{\mu(x),\,\mu(y)} \mu(x) b^{(2)}_{x,y}(\underline{\mu},\underline{\ell})\; P\left[\mu(x),t \mid \ell(x)\right] P\left[\mu(y),t \mid \ell(y)\right] \qquad \text{if } x \neq y$$

$$\sum_{\mu(y)} \mu(y) b^{(2)}_{y,y}(\underline{\mu},\underline{\ell})\; P\left[\mu(y),t \mid \ell(y)\right] \qquad \text{if } x = y,$$

$$H_{\bar{B}}(t|x) = \sum_{i} H_{\bar{B}}(t|x_i), \tag{3.4-4}$$

$$H_{\bar{B}}(t|x_i) = \int_{o}^{t} \lambda(u \mid x_i)\; r(u \mid x)\; P_{\bar{B}}(t-u \mid x_i)\; du, \tag{3.4-5}$$

$$H_{L}(t|x) = \sum_{i} H_{L}(t|x_i), \tag{3.4-6}$$

$$H_{L}(t|x_i) = \int_{o}^{t} \lambda(u \mid x_i)\; r(u \mid x)\; P_{L}(t-u \mid x_i)\; du \tag{3.4-7}$$

where $P_{\bar{B}}$ and P_{L} are both obtained from the Single Fault Coverage Model for fault category x_i,

$$P(\mu(x),t \mid \ell(x)) = \binom{\ell(x)}{\mu(x)} \left[a(t|x)\right]^{\mu(x)} \left[1-a(t|x)\right]^{\ell(x)-\mu(x)}, \tag{3.4-8}$$

$$a(t|x) = H_{L}(t|x) \Big/ 1 - r(t|x)\,, \tag{3.4-9}$$

$$r(t|x) = \exp\left[-\int_0^t \sum_i \lambda(u \mid x_i)\, du\right]. \qquad (3.4\text{-}10)$$

$b^{(2)}_{x,y}(\underline{\mu}, \underline{\ell})$ is defined by (3.4-11)

$$\sum_i \frac{\binom{\mu(x)}{i} \binom{n(x)-\ell(x)}{q(x)-i}}{\binom{q(x)}{i} \binom{n(x)-\ell(x)+\mu(x)}{q(x)} \binom{n(y)-\ell(y)+\mu(y)\,\delta(x,y)}{}}$$

$$\cdot \sum_{j=1}^{i} (-1)^{j-1} N\left(q(x), q(y) \mid j\right) \left[1+\delta(x,y)\,\delta(j,1)\right]$$

$$\cdot \binom{q(x)-j-\delta(x,y)}{i-j} \quad ,$$

$N\left(q(x),q(y) \mid j\right)$ = number of sets of j critical pairs that couple a fixed y module, among first q(y) modules, with j distinct stage-x modules, among first q(x), (3.4-12)

$q(x)$ = $q(i,x)$ if $NOP(x) = (q(1,x), q(2,x), \ldots)$ and $q(i-1,x) > n(x) - \ell(x) + \mu(x) \geqslant q(i,x)$. (3.4-13)

(Note: $b^{(2)}_{x,y}(\underline{\mu}, \underline{\ell})$ is not implemented in CARE III, Version 3.

3.4.3 Calculation of $\mu(t)$

Let $(t \mid \underline{\ell})$ denote the rate of a transition at time t from state $G(\underline{\ell})$ to state $F(\underline{\ell})$. As described in Section 3.2.2, two types of changes contribute to such a transition:

(i) a single latent in-use module in error state E propagates without detection (i.e., transition into state F in the corresponding SFCM); or

(ii) a latent in-use critical pair in state AB moves to DF in the corresponding DFCM.

Let $a'(t \mid \underline{\ell})$ and $A'(t \mid \underline{\ell})$ denote respectively the ensemble rates due to each of the two types of changes. It follows then that their sum is an upper bound for the rate $\mu(t \mid \underline{\ell})$.

Using properties P1 and P2 in Section 3.4.1, it follows that $a'(t \mid \underline{\ell})$ is given by the expression

$$\sum_d \sum_x \sum_a a'(t \mid x,a)\, P\left[Y_t = \underline{d};\ (x,a) \text{ in-use latent} \;\middle|\; Y_t = G(\underline{\ell}) \right],$$

where $a'(t \mid (x,a))$ is the rate of a transition $G(\underline{\ell})$ to $F(\underline{\ell})$ due to error propagation in module (x,a).

By conditioning on the category that caused the fault on module (x,a), and on the time of occurrence of the fault one obtains that:

$$a'(t \mid x,a) = \sum_i h_F(t \mid x_i) \Big/ H_L(t \mid x),$$

where $H_L(t \mid x)$ is as defined in Section 3.4.2, and $h_F(t \mid x_i)$ is the rate of error propagation failure due to a category x_i fault.

Since

$$\sum_d \sum_a P\left[Y_t = \underline{d};\ (x,a) \text{ single, in-use, latent} \;\middle|\; Y_t = G(\underline{\ell}) \right]$$
$$\leqslant E\left[\text{number of stage-x latent modules} \;\middle|\; Y_t = G(\underline{\ell}) \right] = \ell(x)a(t \mid x),$$

where $a(t \mid x)$ is defined in Section 3.4.2.

It follows that

$$a'(t \mid \underline{\ell}) \leqslant \sum_{x} \sum_{i} \ell(x) \frac{h_F(t \mid x_i)}{1 - r(t \mid x)}$$

where $h_F(t \mid x_i)$ is evaluated by conditioning on the time of occurrence of the x_i fault.

Similar steps are taken to evaluate $A'(t \mid \underline{\ell})$. First consider all $\underline{d}$ states that form the aggregate state $G(\underline{\ell})$. Then for each state $\underline{d}$ consider all latent critical pairs of faults (x_i, y_j) that can lead to failure of the system. Finally condition on both the time of occurrence of first fault, and on time of occurrence of the second given that at that instant the first fault is benign. These steps lead to the following mathematical expression:

$$A'(t \mid \underline{\ell}) = \sum_{x,i} \sum_{y,j} \frac{h_{DF}(t \mid x_i, y_j)}{H_L(t \mid x)\, H_L(t \mid y)} \quad B(t,(x,y) \mid \underline{\ell}),$$

where

$H_L(t \mid x)$: as defined in Section 3.4.2.,

$h_{DF}(t \mid x_i, y_j)$: rate at which an (x_i, y_j) critical pair causes system failure,

$B(t,(x,y) \mid \underline{\ell})$: expected number of times a given (x,y) pair is latent, in-use and critical at time t, given $\underline{\ell}$ faults.

As was the case for $\lambda^{(2)}(t)$, the evaluation of $B(t,(x,y) \mid \underline{\ell})$ requires detailed analysis of the state space. Use of assumption A1 then leads to the estimate:

$$B(t,(x,y) \mid \underline{\ell}) = \sum_{\substack{\mu(x) \\ \mu(y)}} b^{(1)}_{x,y}(\underline{\mu}, \underline{\ell})\, \mu(x)\, \mu(y)\, P\left[Y_t=(\mu(x), \mu(y)) \mid Y_t=(\ell(x), \ell(y))\right]$$

where

$b^{(1)}_{x,y}(\underline{\mu}, \underline{\ell})$: probability that a given (x,y) pair is critical when chosen from existing latent, in-use stage-x and stage-y modules, given $\underline{\mu}$ latent and $\underline{\ell}$ faulty modules.

The functions h_F, H_B and H_L are evaluated by conditioning on the time of occurrence of the x_i fault. The function h_{DF} is evaluated by conditioning on the time of occurrence of the y_j fault given that the first fault is benign. The function $b^{(1)}_{x,y}$ is obtained by conditioning on the number of latent in-use modules in stages x and y.

In summary, the rate for a transition from $G(\underline{\ell})$ into $F(\underline{\ell})$ is given by

$$\mu(t \mid \ell) = a'(t \mid \underline{\ell}) + A'(t \mid \underline{\ell}), \tag{3.4-14}$$

where

$$a'(t \mid \underline{\ell}) = \sum_{x,i} \ell(x)\, \frac{h_F(t \mid x_i)}{1-r(t \mid x)}, \tag{3.4-15}$$

$$A'(t \mid \underline{\ell}) = \sum_{x,i} \sum_{y,j} \frac{h_{DF}(t \mid x_i, y_j)}{H_L(t \mid x)\, H_L(t \mid y)}\, B(t,(x,y) \mid \underline{\ell}), \tag{3.4-16}$$

$$h_F(t \mid x_i) = \int_0^t \lambda(u \mid x_i)\, r(u \mid x_i)\, p_F(t-u \mid x_i)\,du, \tag{3.4-15}$$

$$r(t \mid x) = \exp\left[-\int_0^t \sum_i \lambda(u \mid x_i)\, du\right], \tag{3.4-16}$$

$$h_{DF}(t \mid x_i y_j) = \int_0^t H_B(u \mid x_i)\lambda(u \mid y_j)\, r(u \mid y)\, p_{DF}(t-u \mid x_i y_j)\,du, \tag{3.4-17}$$

$$H_B(t \mid x_i) = \int_0^t \lambda(u \mid x_i)\, r(u \mid x)\, P_B(t-u \mid x_i)\, du, \tag{3.4-18}$$

$$H_L(t \mid x) = \sum_i \int_0^t \lambda(u \mid x_i)\, r(u \mid x)\, P_L(t-u \mid x_i)\, du, \tag{3.4-19}$$

$B(t,(x,y)\mid \underline{\ell})$ as defined by (3.4-20)

$$\sum_{\substack{\mu(x)\\ \mu(y)}} b^{(1)}_{x,y}(\underline{\mu},\underline{\ell})\, \mu(x)\, \mu(y)\, P(\mu(x),t \mid \ell(x))\, P(\mu(y),t \mid \ell(y)) \qquad \text{if } x \neq y$$

$$\sum_{\mu(y)} b^{(1)}_{y,y}(\underline{\mu},\underline{\ell})\, \mu(y)\, (\mu(y)-1)\, P(\mu(y),t \mid \ell(y)) \qquad \text{if } x = y,$$

$$P(\mu(x),t \mid \ell(x)) = \binom{\ell(x)}{\mu(x)} \left[a(t \mid x)\right]^{\mu(x)} \left[1-a(t \mid x)\right]^{\ell(x)-\mu(x)}, \tag{3.4-21}$$

$$a(t \mid x) = H_L(t \mid x) / 1 - r(t \mid x), \tag{3.4-22}$$

$b^{(1)}_{x,y}(\underline{\mu}, \underline{\ell})$ is defined by (3.4-22)

$$\frac{N(q(x),q(y)|1)}{q(x)q(y)} \left[1 - \frac{\binom{n(x)-\ell(x)}{q(x)}}{\binom{n(x)-\ell(x)+\mu(x)}{q(x)}}\right] \left[1 - \frac{\binom{n(y)-\ell(y)}{q(y)}}{\binom{n(y)-\ell(y)+\mu(y)}{q(y)}}\right] \quad \text{if } x \neq y$$

$$\frac{2N(q(y),q(y)|1)}{q(y)(q(y)-1)} \left[1 - \frac{\binom{n(y)-\ell(y)}{q(y)} + \mu(y)\binom{n(y)-\ell(y)}{q(y)-1}}{\binom{n(y)-\ell(y)+\mu(y)}{q(y)}}\right] \quad \text{if } x = y,$$

$N(q(x),q(y)|1))$ = number of (x,y) critical pairs among first q(x) and q(y) modules, (3.4-23)

$q(x)$ = $q(i,x)$ if $NOP(x) = (q(1,x), q(2,x),....)$ and $q(i-1,x) > n(x) - \ell(x) + \mu(x) \geq q(i,x)$. (3.4-24)

(Note: $b^{(1)}_{x,y}(\underline{u}, \underline{\ell})$ is not implemented in CARE III, Version 3).

3.5 IMPLEMENTATION IN CARE-III

In the previous sections the formulation of the reliability model was reviewed. The objective of this section is to document the model that is actually implemented and outline the calculations performed. In the following sections, the overall structure and data flow of the CARE3 program are described in some detail, the solution of the reliability model is outlined and the basic reliability functions are defined.

3.5.1 Overview of CARE3 Program

Figure 3.5-1 illustrates the overall data flow for the CARE3 program. The user's input data for the reliability model is read from file CREIN by the input program CAREIN. It includes stage data (parameters N, M, NOP, LC), fault category data for each stage (parameters type, ω, λ), the system fault tree and any critical pairs fault trees. After the data is checked and preprocessed by CAREIN, it is passed to CARE3 on files RELIN, FT15F and BXYIN. The moments of the coverage functions are also passed to CARE3 on file CVGMTS. The reliability model is solved by CARE3 and the functions P^{*}_{SUM} and Q_{SUM} are computed. In addition the reliability functions are passed to the plotting program, RELPLT, on file PLTFL.

Figure 3.5-2 provides a high level functional description of the CARE3 program; a brief explanation of the functions follows:

- Computation Control

  These subroutines control the computations for solving the reliability model; the details of the computational sequence are given in Section 3.5.2 to 3.5.5. Figures 3.5-3, -4 and -5 illustrate the control structure of subroutines CARE3, RLSBRN, NFLTVDP and GNFLTVC with call trees.

- Reliability Functions

  The subroutines in this group compute the basic reliability functions used in the solution of the reliability model; definitions of the functions are given in Section 3.5.5.

o Numerical Integration

  The subroutines in this group are used to compute numerically the integral (over a time interval) of a function. This kind of calculation is required in the solution of the integral equation for the functions $Q(t|\underline{\ell})$. The numerical methods used in these subroutines are discussed in detail in Section 3.6.

- Numerical Convolution

  The subroutines in this group are used to compute numerically the convolution of two functions. This kind of calculation is required to compute the (time dependent) transition rates of the reliability model for the situation of non-perfect coverage. These are the most crucial numerical subroutines because they are part of the interface between the coverage and reliability models; the numerical methods used are discussed in detail in Section 3.6.

- Support Functions

  These are "library" type subroutines which are used by all other subroutines in the program for very basic operations or calculations.

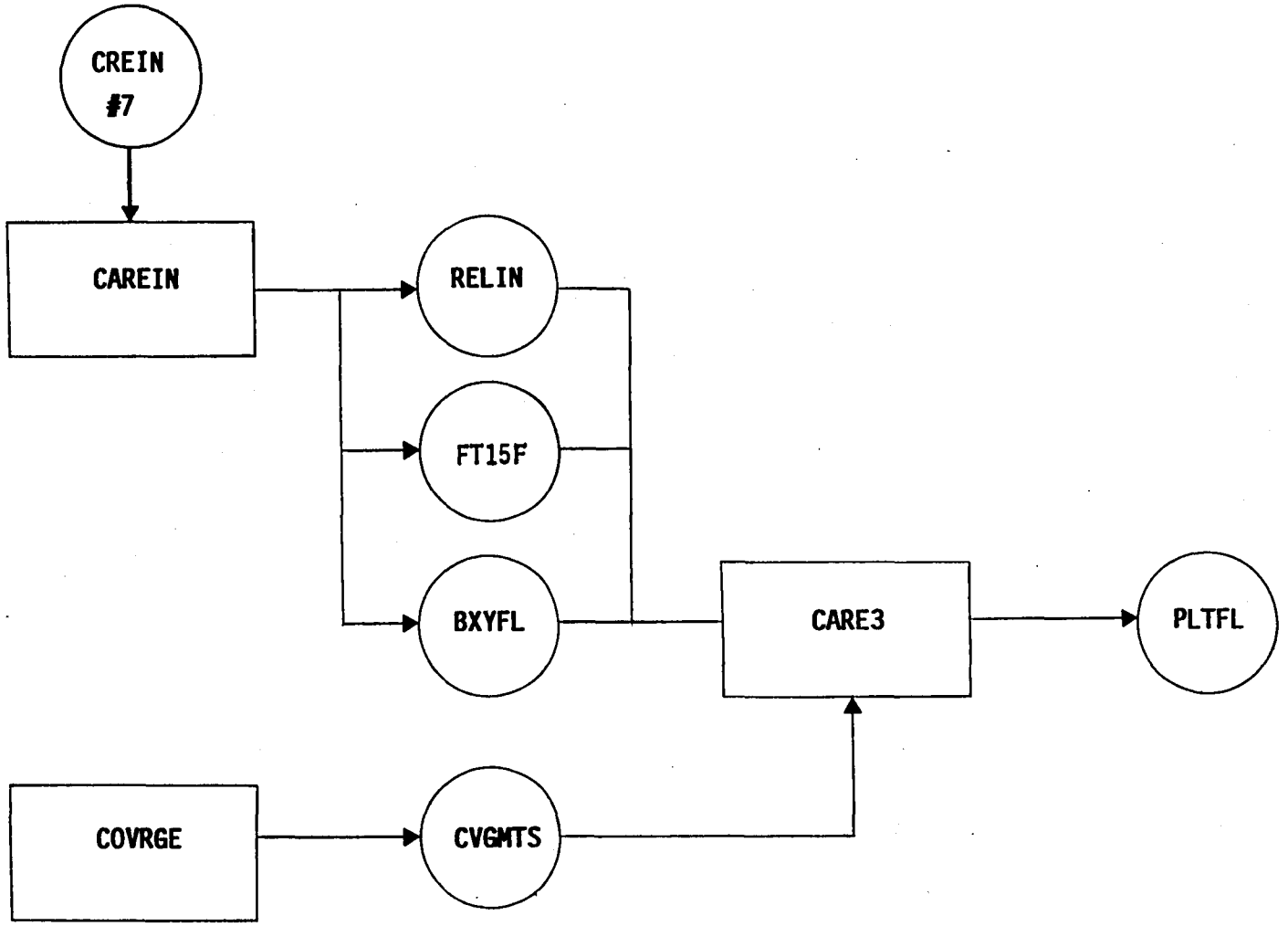


FIGURE 3.5-1 Data Flow for CARE3 Program

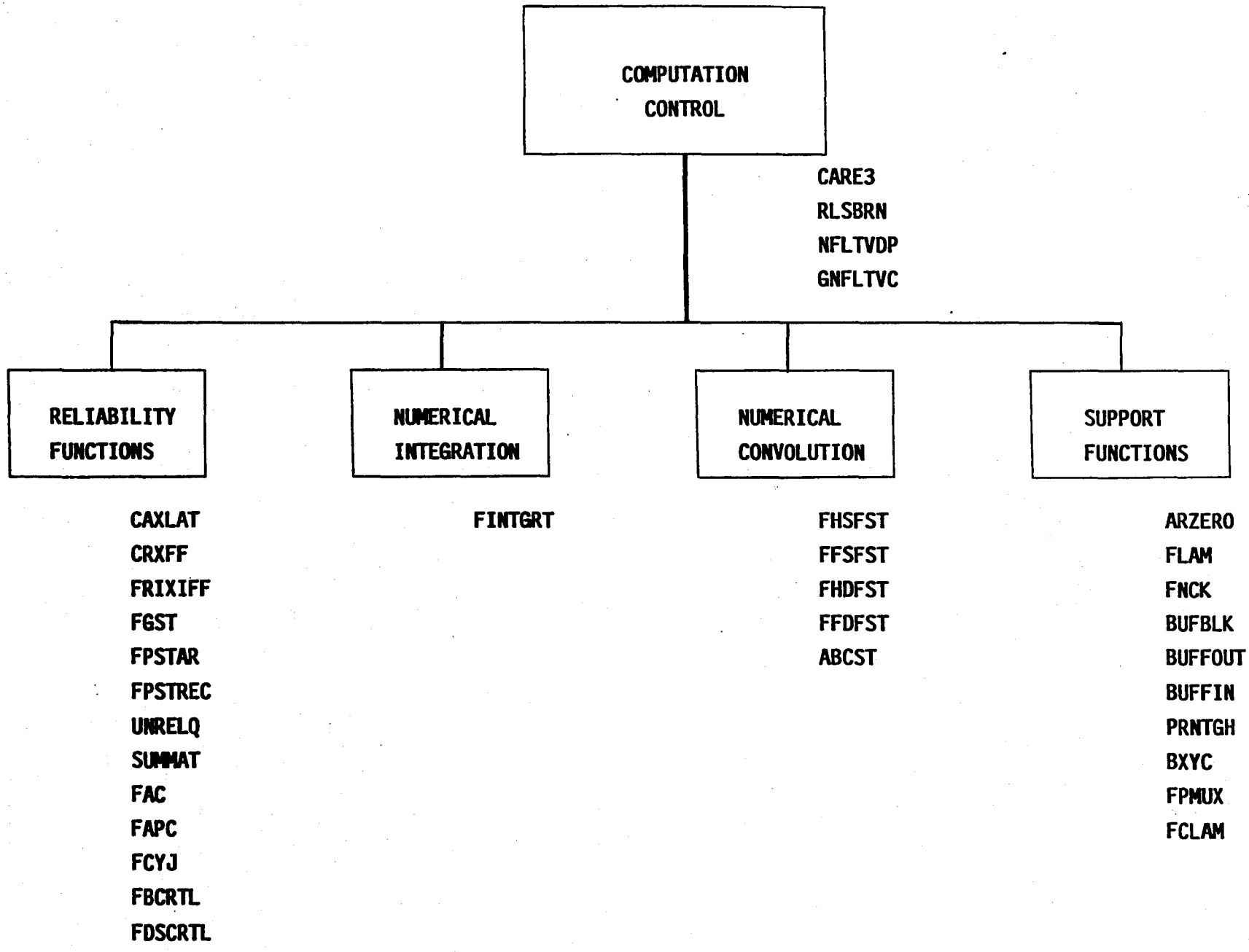


FIGURE 3.5-2 Functional Structure of CARE3 Program

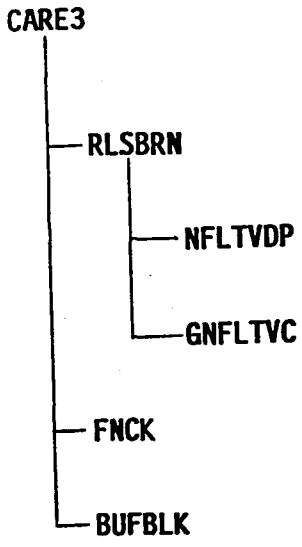


FIGURE 3.5-3 CARE3 Call Tree

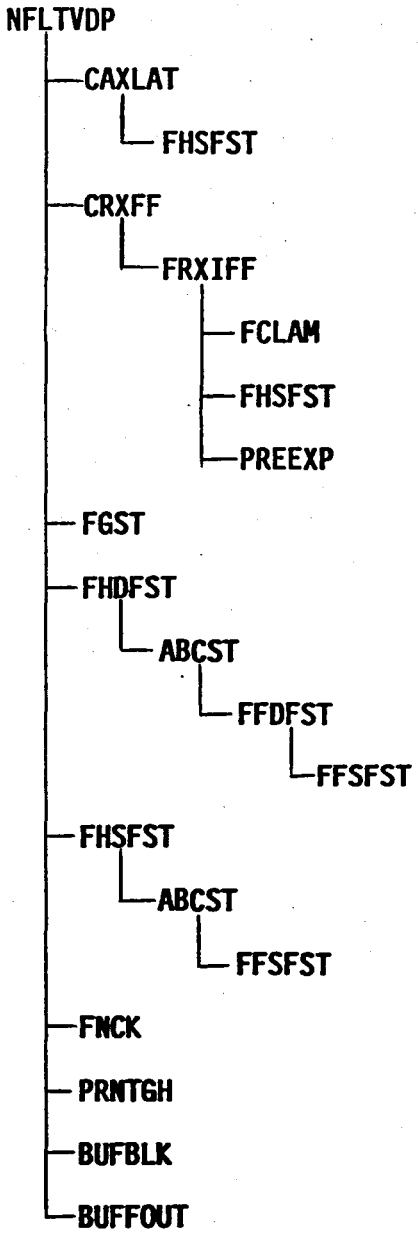


FIGURE 3.5-4 NFLTVDP Call Tree

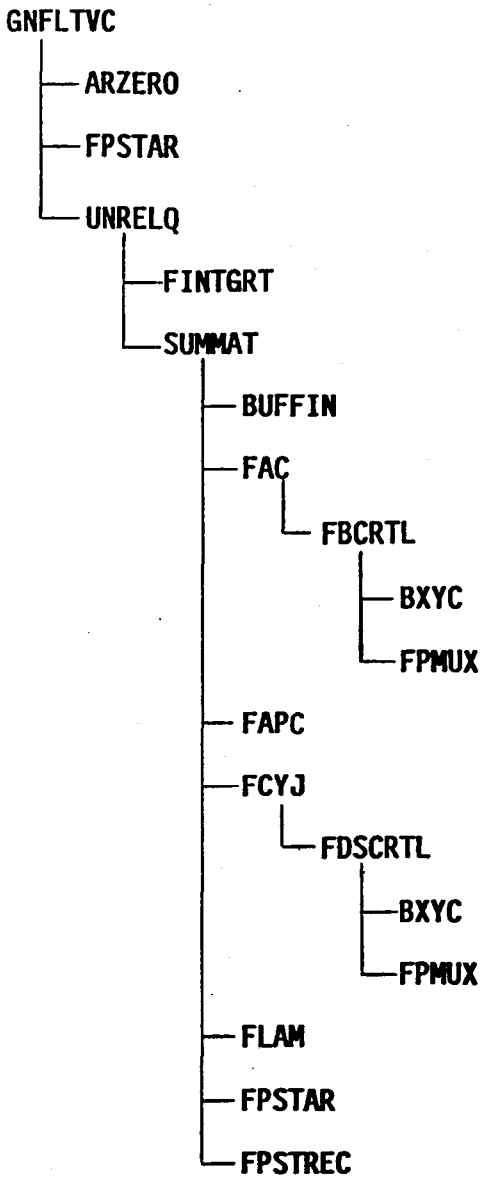


FIGURE 3.5-5 GNFLTVC Call Tree

3.5.2 System Fault Tree

The preprocessing of the system fault tree is performed by the input program CAREIN. Figure 3.5-6 illustrates the user's input data for the system fault tree (on file CREIN) and the corresponding input for FTREE generated by CAREIN. (Refer to the CARE-III User's Manual for a description of the input data formats.) A brief description of each of the parameters in Figure 3.5-6 follows.

- TITLE: One or more lines of descriptive text.
- IROP: FTREE run option; set to 3 by CAREIN.
- MCOMB: The maximum number of input gate (stage) failures; set to the maximum of 4 and KWT by CAREIN, where KWT is described in the text below.
- PSTRNC: Perfect coverage truncation value; set to 10^{-14} by CAREIN.
- IFSTG: The number of the first input gate (stage); it must be set to 1 by the user.
- ILSTG: The number of the last input gate (stage); it must be set to NSTGES by the user.
- IFGTE: The number of the first logic gate in the system fault tree.
- ILGTE: The number of the last logic gate in the system fault tree.
- FTHRS: The flight time (in hours); computed by CAREIN from the input variable FT in NAMELIST/RNTIME/.

- ISTG: The number of an input gate (stage); CAREIN generates an input gate for each stage, i.e., ISTG ranges from 1 to NSTGES.

- EFCTLM: The failure rate of an input gate (stage); the CAREIN estimate of the failure rate for a stage is described in the text below.

- I1: FTREE control code set to 1 by CAREIN: this forces FTREE to consider EFCTLM the failure rate and FTHRS the time used to compute the probability of failure for the input gate (stage).

- IGTE: The number of a logic gate in the system fault tree.

- ITYP: The type of a logic gate; refer to the CARE-III User's Manual for a description of the logic gate types.

- INPUTS: A string of gate numbers listing the inputs to the logic gate.

Figure 3.5-7 illustrates the output data (MINTRM) file generated on unit FT15F when the system fault tree is processed by FTREE. A brief description of each of the parameters in Figure 3.5-7 follows.

- PRBMT: The FTREE estimate of the probability of failure of the system (after a flight time of FTHR) due to the set of stage failures indicated by the corresponding MINTRM.

- MINTRM: A fault vector (1 = failed, 0 = not-failed) for a failed state of the system; each fault vector has NSTGES components (i.e., one for each stage).

DATA TYPE	USER'S INPUT FILE(CREIN)	FTREE INPUT FILE(FT11F)
TITLE AND CONTROL BLOCK	TITLE * IFSTG,ILSTG,IFGTE,ILGTE ** FTHRS	TITLE IROP,MCOMB,PSTRNC IFSTG,ILSTG,IFGTE,ILGTE ** FTHRS
INPUT BLOCK	. . . * . . .	. . . ISTG,EFCTLM,I1 . . .
LOGIC BLOCK	. . . IGTE,ITYP,INPUTS TO GATE . . .	. . . IGTE,ITYP,INPUTS TO GATE . . .

* Record generated by CAREIN
** Record not used by CARE3 version of FTREE

FIGURE 3.5-6 System Fault Tree: FTREE Input

PRBMT, MINTRM

PRBMT, MINTRM

PRBMT, MINTRM

FIGURE 3.5-7 System Fault Tree: FTREE Output

3.5.3 Critical Pairs Fault Tree(s)

The preprocessing of the critical pairs fault tree(s) is performed by the input program CAREIN. Figure 3.5-8 illustrates the user's input data for a critical pairs fault tree (on file CREIN) and the corresponding input for FTREE generated by CAREIN. (Refer to the CARE-III User's Manual for a description of the input data formats.) A brief description of each of the parameters in Figure 3.5-8 follows:

- TITLE: One or more lines of descriptive text.

- IROP: FTREE run option; set to 3 by CAREIN.

- MCOMB: The maximum number of input gate (unit) failures; set to 2 by CAREIN.

- PSTRNC: Perfect coverage truncation value; set to 10^{-14} by CAREIN.

- IFUNT: The number of the first input gate (unit); it must be set by user to the number of the first unit in the first stage of the tree.

- ILUNT: The number of the last input gate (unit); it must be set by user to the number of the last unit in the last stage of the tree.

- IFGTE: The number of the first logic gate in the critical pairs fault tree.

- ILGTE: The number of the last logic gate in the critical pairs fault tree.

- FTHRS: The flight time (in hours); computed by CAREIN from the input variable FT in NAMELIST/RNTIME/.

- ISTG: The number of a stage in the critical pairs tree; CAREIN generates (ISLUNT-ISFUNT + 1) FTREE input gates (modules), for each stage.

- ISFUNT: The number of the first input gate (module) in the stage numbered ISTG.

- ISLUNT: The number of the last input gate (module) in the stage numbered ISTG.

- IUNT: The number of an input gate (module) in the stage numbered ISTG: ISFUNT IUNT ISLUNT.

- UNTLM: The failure of an input gate (module) in the stage numbered ISTG; the CAREIN estimate of the failure rate for the units in a stage is described in the text below.

- T1: FTREE control code set to 1 by CAREIN: this forces FTREE to consider UNTLM the failure rate and FTHRS the time used to compute the probability of failure of the input gate (module).

- IGTE: The number of a logic gate in the critical pairs fault tree.

- ITYP: The type of a logic gate; refer to the CARE-III User's Manual for a description of the logic gate types.

- INPUTS: A string of gate numbers listing the inputs to the logic gate.

Figure 3.5-9 illustrates the output data (MINTRM) file generated on unit FT25F when a critical pairs fault tree is processed by FTREE. Such a set of records is written to unit FT25F for each of the critical pairs trees input by the user. A brief description of each of the parameters in Figure 3.5-9 follows.

- NUNTS: The total number of modules in the stages in the critical pairs trees (ILUNT-IFUNT + 1)

- PRBMT: The FTREE estimate of the probability of a critical pair system failure due to the set of unit failures indicated by the corresponding MINTRM.

- MINTRM: A fault vector (1 = failed, 0 = not-failed) for a failed state of the system; each fault vector has NUNTS components (i.e., one for each unit).

In Section 3.4, it was shown that the critical pairs fault tree data is used in the calculation of the rate for transitions between the $Q(\underline{\ell})$ and $F(\underline{\ell})$ or $F(\underline{\ell}+\underline{1}(y))$ states. In particular the factors:

$$b^{(1)}_{x,y}(\underline{\mu},\underline{\ell}),$$

$$b^{(2)}_{x,y}(\underline{\mu},\underline{\ell}),$$

depend on the number of (x,y) critical pairs, N(x,y). However, in the CARE-III program, the corresponding calculations involve the factor:

$$b_{xy}(\ell(x)-\mu(x),\ \ell(y)-\mu(y))$$

which is defined verbally on page 6 of the CARE-III Maintenance Manual, L. A. Bryant, and J. J. Stiffler (1982a), but is not defined by an equation in any CARE-III document. Review of subroutine CRTLPRS indicates the following definition:

$$b_{xy}(\ell(x)-\mu(x),\ \ell(y)-\mu(y)) = \begin{cases} \dfrac{k_{xy}(\ell(x)-\mu(x),\ \ell(y)-\mu(y))}{(n(x)-\ell(x)+\mu(x))(n(y)-\ell(y)+\mu(y))} & : x \neq y, \\ \dfrac{k_{xy}(\ell(x)-\mu(x),\ \ell(x)-\mu(x))}{(n(x)-\ell(x)+\mu(x))(n(x)-\ell(x)+\mu(x)-1)} & : x = y, \end{cases}$$

where $k_{xy}(\ell(x)-\mu(x),\ \ell(y)-\mu(y))$ is the number of (x,y) critical pairs (x,a), (y,b) such that there are at least $\ell(x)-\mu(x)$ modules (x,a') and at least $\ell(y)-\mu(y)$ modules (y,b') for which

$$a < a',$$
$$b < b'.$$

Note that, since the counts k_{xy} depend on the numbers assigned to the modules, the b_{xy} and hence the solution of the reliability model will depend on the numbering of the modules. This situation is inconsistent with the assumption that all modules within a stage are identical. The calculation of k_{xy} is further complicated by logic which depends on the (user supplied) NOP data; BCS has not been able to make a reasonable interpretation of this logic. The (user supplied) LC data is available, but not used, in the k_{xy} calcualation.

DATA TYPE	USER'S INPUT FILE(CREIN)	FTREE INPUT FILE(FT11F)
TITLE AND CONTROL BLOCK	TITLE * IFUNT,ILUNT,IFGTE,ILGTE ** FTHRS	TITLE IROP,MCOMB,PSTRNC IFUNT,ILUNT,IFGTE,ILGTE ** FTHRS
INPUT BLOCK	. . . +ISTG,ISFUNT,ISLUNT . . .	. . . IUNT,UNTLM,I1 . . .
LOGIC BLOCK	. . . IGTE,ITYP,INPUTS TO GATE . . .	. . . IGTE,ITYP,INPUTS TO GATE . . .

* Record generated by CAREIN
** Record not used by CARE3 version of FTREE
\+ Not a standard FTREE input block record;
(ISLUNT-ISFUNT+1)FTREE input block records generated per second

FIGURE 3.5-8 Critical Pairs Fault Tree: FTREE Input

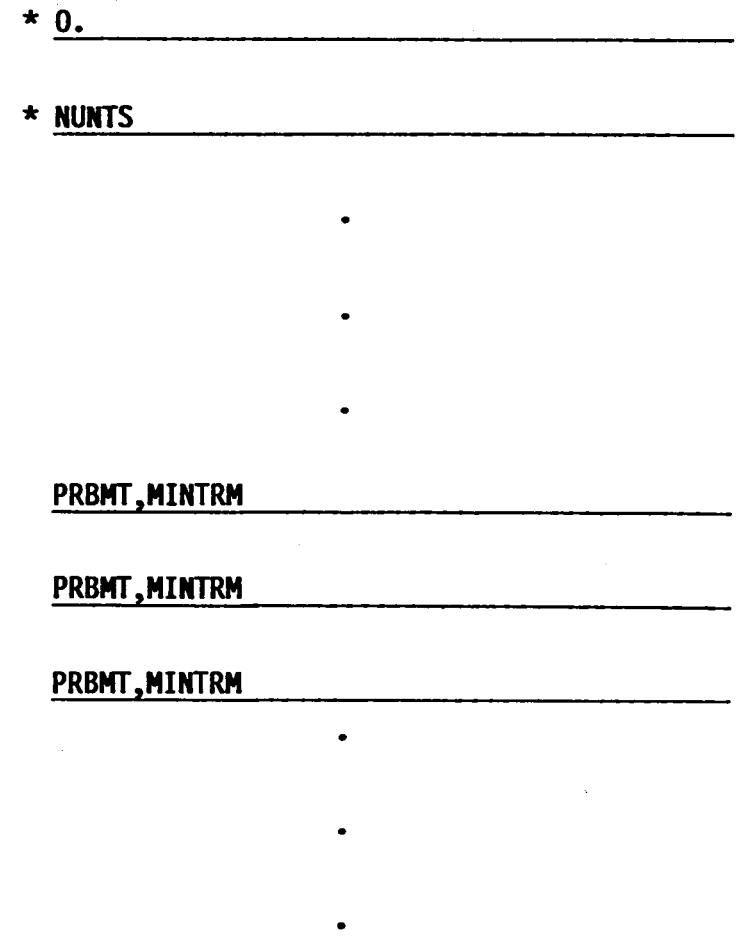


FIGURE 3.5-9 Critical Pairs Fault Tree: FTREE Output

3.5.4 Outline of Calculations

The evaluation of the system reliability R(t) is performed by the main program CARE3; the calculation is partitioned into "SUBRUN's" which consist of the evaluation of the reliability of subsystems which are independent in the sense that modules in different subsystems are not critically coupled as defined by the critical pairs tree(s). (The possible coupling of subsystems by the system fault tree does not appear to be considered.) For each SUBRUN, the calculation of reliability is controlled by RLSBRN which calls NFLTVDP and GNFLTVC to generate all fault vectors $\underline{\ell}$ for the subsystem, partition the fault vectors into two disjoint subsets L_s and $\bar{L}_s$ and compute $Q(t \mid \underline{\ell})$ for $\underline{\ell} \in L_s$ and $P^*(t \mid \underline{\ell})$ for $\underline{\ell} \in \bar{L}_s$. The calculations performed by NFLTVDP and GNFLTVC are outlined in Tables 3.5-1 and 2, respectively, and all basic reliability functions used in the calculations are defined in Section 3.5.5.

The calculation of the system unreliability as a function of SUBRUN results is implemented in GNFLTVC and CARE3 and depends on whether or not the user supplies a system fault tree. BCS has carefully reviewed the program logic and determined that the following equations define the calculation of system unreliability that is actually implemented in CARE-III.

- No System Fault Tree

In this case, the system unreliability is computed by GNFLTVC as follows:

$$1.-R(t) = \sum_{\text{SUBRUN's}} \left[\sum_{\underline{\ell} \in L_s} Q(t \mid \underline{\ell}) + \sum_{\underline{\ell} \in \bar{L}_s} P^*(t \mid \underline{\ell}) \right], \qquad 3.5\text{-}1$$

where

$$Q(t \mid \underline{\ell}) = \int_0^t K(t \mid \underline{\ell})\, d\tau, \qquad 3.5\text{-}2$$

$$P^*(t \mid \underline{\ell}) = \prod_{x \in \text{SUBRUN}} \binom{n(x)}{\ell(x)} (r(t|x))^{n(x)-\ell(x)} (1.-r(t|x))^{\ell(x)} . \qquad 3.5\text{-}3$$

Under the assumption that the default system tree is an OR tree (spares exhaustion of any stage fails the system), equation 3.5-1 reduces to equation 3.3-5 for the case of one SUBRUN. For the case of more than one SUBRUN, the interpelation of equation 3.5-1 and its relation to equation 3.3-5 is no longer clear.

- System Fault Tree

In this case, the system unreliability is computed by GNFLTVC and CARE3 as follows:

$$1.-R(t) = \sum_{\text{SUBRUN's}} \left[\sum_{\underline{\ell} \in L_s} Q(t \mid \underline{\ell}) \right] + \sum_{\text{MINTRM's}} \left[\prod_{x} \begin{cases} P^*(t|x) & :\min(x)=1 \\ 1.-P^*(t|x) & :\min(x)=0 \end{cases} \right] \qquad 3.5.4$$

where

$$P^*(t \mid x) = \sum_{\ell(x)=n(x)-m(x)+1}^{n(x)} \binom{n(x)}{\ell(x)} (r(t|x))^{n(x)-\ell(x)} (1.-r(t|x))^{\ell(x)} , \qquad 3.5.5$$

and the MINTRM's in the second term of equation 3.5-4 are defined by FTREE from the system fault tree. The second term in equation 3.5-4 appears to be a correct interpretation of the FTREE output MINTRM file and a good approximation to the second term in equation 3.3-5. However, for the first term, the relation between L_s for a SUBRUN and the system fault tree (i.e., L in equation 3.3-5) is not clear and appears to be in error even for the case of one SUBRUN. For the case of more than one SUBRUN, the

interpretation of the first term in equation 3.5-1 and its relation to the first term in equation 3.3-5 is no longer clear.

BCS is currently investigating the questions raised by these observations.

Table 3.5-1
RELIABILITY CALCULATIONS (Non-ℓ-dependent)

FUNCTION	DESCRIPTION	SUBROUTINE	ARRAYS	PARAMETERS
$\lambda(t\|x_i)$	Rate of occurrence of category $-x_i$ faults at time t.	FLAM	-	$\lambda(x_i)$ $\omega(x_i)$
$\Lambda(t\|x_i)$	Cumulative rate of occurrence of category $-x_i$ faults.	FCLAM	-	$\lambda(t\|x_i)$
$r(t\|x_i)$	Probability that a stage -x module has not experienced at category $-x_i$ fault by time t.	FRXIFF	-	$\Lambda(t\|x_i)$ $h_{DPT}(t\|x_i)$
$r(t\|x)$	Reliability of a stage -x module at time t.	CRXFF	RXAR	$r(t\|x_i)$
$a(t\|x_i)$	Probability that a stage -x module has a latent, non-transient (transient) category $-x_i$ fault at time t, given that it has (not) experienced a non-transient or leaky transient fault by time t.	CAXLAT	AXIAR	$H_L(t\|x_i)$ $r(t\|x)$

Table 3.5-1 (Continued)
RELIABILITY CALCULATIONS (Non-ℓ-dependent)

FUNCTION	DESCRIPTION	SUBROUTINE	ARRAYS	PARAMETERS
$a(t\|x)$	Probability that a stage -x module has a latent, non-transient fault at time t, given that it has experienced some non-transient fault at time t.	FDSCRTL FBCRTL		$a(t\|x_i)$
$h_{DPT}(t\|x_i)$	Rate at which a transient, category $-x_i$ fault is detected as permanent at time t.	FHSFST	PDP	a_{DP}, b_{DP}, c_{DP} $m_{DF}^0, m_{DF}^1, m_{DF}^2$
$H_L(t\|x_i)$	Probability of a latent, category $-x_i$ fault at time t.	FHSFST	PLAT (TAPE9)	a_L, b_L, c_L M_L^0, M_L^1, M_L^2
$h_F(t\|x_i)$	Rate of error propagation system failure due to a category $-x_i$ fault at time t.	FHSFST	GORHSF	a_F, b_F, c_F m_F^0, m_F^1, m_F^2
$g_F(t\|x_i)$	Rate of error propagation system failure due to a category $-x_i$ fault at time t, given that it was latent prior to t.	FGST	GORHSF (TAPE10)	$h_F(t\|x_i)$ $a(t\|x_i)$

Table 3.5-1 (Continued)
RELIABILITY CALCULATIONS (Non-ℓ-dependent)

FUNCTION	DESCRIPTION	SUBROUTINE	ARRAYS	PARAMETERS
$H_{\overline{B}}(t\|x_i)$	Probability of a non-benign, latent category $-x_i$ fault at time t.	FHSFST	PNBNG	$a_{\overline{B}}, b_{\overline{B}}, c_{\overline{B}}$ $M^0_{\overline{B}}, M^1_{\overline{B}}, M^2_{\overline{B}}$
$G_{\overline{B}}(t\|x_i)$	Probability of a non-benign, latent category $-x_i$ fault at time t, given that it was latent prior to t.	FGST	GORHSF (TAPE11)	$H_{\overline{B}}(t\|x_i)$ $a(t\|x_i)$
$H_B(t\|x_i)$	Probability of a benign, latent category $-x_i$ fault at time t.	FHSFST	PBNG	a_B, b_B, c_B M^0_B, M^1_B, M^2_B
$h_{DF}(t\|x_i)$	Rate of system failure due to critically coupled category $-x_i$ and category $-y_j$ faults at time t.	FHDFST	HDFPTS (TAPE12)	a_{DF}, b_{DF}, c_{DF} $m^0_{DF}, m^1_{DF}, m^2_{DF}$

Table 3.5-2
RELIABILITY CALCULATIONS ($\underline{\ell}$-dependent)

FUNCTION	DESCRIPTION	SUBROUTINE	ARRAYS	PARAMETERS
$P^*(t\|\underline{\ell})$	Probability that a system has sustained exactly $\underline{\ell}$ failures by time t.	FPSTAR FPSTREE	-	$\underline{n}, \underline{\ell}$ $r(t\|x)$
$P(\mu(x),t\|\ell(x))$	Probability that a system has $\mu(x)$ stage -x latent, permanent faults given that it has $\ell(x)$ faults.	FPMUX	-	$\underline{\mu}, \underline{\ell}$ $a(t\|x)$
$K(t\|\underline{\ell})$		SUMMAT	SUMK	$c(t\|\underline{\ell},y_j)$ $P^*(t\|\underline{\ell})$ $A(t\|\underline{\ell})$ $a'(t\|\underline{\ell})$
$C(t\|\underline{\ell},y_j)$	Probability of system failure due to a category $-y_j$ fault at time t, given $\underline{\ell}$ faults at time t.	FCYJ	GNBNG	$G_B^-(t\|x_i)$ $D(t\|\underline{\ell},x_i,y_j)$

Table 3.5-2 (Continued)
RELIABILITY CALCULATIONS ($\underline{\ell}$-dependent)

FUNCTION	DESCRIPTION	SUBROUTINE	ARRAYS	PARAMETERS
$A'(t\|\underline{\ell})$	Rate of system failure due to critical fault conditions for $\underline{\ell}$ faults at time t.	FAC	HDFPTS HLAT	$h_{DF}(t\|x_i,y_j)$ $H_L(t\|x_i)$ $B(t\|\underline{\ell},x_i,y_j)$
$a'(t\|\underline{\ell})$	Rate of system failure due to error propagation for $\underline{\ell}$ faults at time t.	FAPC	GFLD	$\underline{n},\underline{\ell}$ $g_F(t\|x_i)$ $a(t\|x_i)$
$D(t\|\underline{\ell},x_i,y_j)$	Expected number of category -x_i, -y_j critical faults, given $\underline{\ell}$ faults at time t that would be created as the result of a stage -y fault at time t.	FDSCRTL	BXYAR	$a(t\|x_i)$ $a(t\|x)$ $P(\mu(x),t\|\ell(x))$ bxy
$B(t\|\underline{\ell},x_i,y_j)$	Expected number of category -x_i, -y_j critical faults, given $\underline{\ell}$ faults at time t.	FBCRTL	BXYAR	$a(t\|x_i)$ $P(\mu(x),t\|\ell(x))$ b_{xky}

3.5.5 Basic Reliability Functions

The basic reliability functions are the time dependent rate and probability functions used in the calculation of the reliability for a subsystem (i.e., a CARE-III SUBRUN). The subroutines, arrays or I/O units used to compute and store the function values were specified in Tables 3.5-1 and -2. In this section the definition of each function, as obtained from the CARE-III code, are presented along with remarks about any inconsistencies with the CARE-III documents and/or BCS' analysis of the CARE-III reliability model. In the definitions, the functions are defined for any time $t>0$, but in the CARE3 program the functions are evaluated only at the discrete time points for which the reliability model is solved:

$$t_j = (j-1)\ \Delta t;\ j = 1,2,\ldots,j\ \text{max.}$$

- $$\lambda(t|x_i) = \omega(x_i)\ \lambda(x_i)^{\omega(x_i)}\, t^{\omega(x_i)-1} \qquad 3.5\text{-}1$$

- $$\Lambda(t|x_i) = \int_0^t \lambda(\tau|x_i)\, d\tau \qquad 3.5\text{-}2$$

$$= \left[\lambda(x_i)t\right]^{\omega(x_i)} \qquad 3.5\text{-}3$$

The CARE-III documents give different definitions for $\lambda(t|x_i)$ and $\Lambda(t|x_i)$; although the documented definitions are consistent with each other, the user will not obtain the Weibull failure model he expects.

- $$r(t|x_i) = \begin{cases} \epsilon - \Lambda(t|x_i) & : x_i \text{ non-transient fault} \\ \epsilon - \int_0^t h_{DPT}(\tau|x_i)d\tau & : x_i \text{ transient fault} \end{cases} \qquad 3.5\text{-}4$$

- $$r(t|x) = \prod_i r(t|x_i)$$

$$\bullet \quad a(t|x_i) = H_L(t|x_i) \left\{ \begin{array}{ll} \dfrac{1.}{1.-r(t|x)} & : x_i \text{ non-transient fault} \\ 1. & : x_i \text{ transient fault} \end{array} \right\} \qquad 3.5\text{-}5$$

$$\bullet \quad a(t|x) = \sum_i a(t|x_i) \qquad 3.5\text{-}6$$

The function $a(t|x)$ is computed as defined in equation 3.5-6 in subroutines FDSCRTL and FBCRTL to compute $D(t|\underline{\ell},x_i,y_j)$ and $B(t|\underline{\ell},x_iy_j)$ and subroutine FPMUX to compute $P(\mu(x),t\,\ell(x))$. BCS has determined that the sum in equation 3.5-6 should be over only non-transient, category $-x_i$ faults to make $a(t|x)$ consistent with its definition and use in CARE-III.

- $P^*(t|\underline{\ell}) = \prod_x \binom{n(x)}{\ell(x)} \left(1.-r(t|x)\right)^{\ell(x)} \left(r(t|x)\right)^{n(x)-\ell(x)}$ 3.5-7

- $P\left(\mu(x),t|\underline{\ell}(x)\right) = \binom{\ell(x)}{\mu(x)} \left(1.-a(t|x)\right)^{\ell(x)-\mu(x)} \left(a(t|x)\right)^{\mu(x)}$ 3.5-8

- $h_{DPT}(t|x_i) = \int_0^t P_{DP}\left(\tau|k(x_i)\right)\lambda\,(t-\tau\,|\,x_i)\,d\tau$ 3.5-9

$= a_{DP}(t|x_i)\, m_{DP}^{o}\left(t|k(x_i)\right) + b_{DP}(t|x_i) m_{DP}^{1}\left(t|k(x_i)\right) + C_{DP}(t|x_i) m_{DP}^{2}\left(t|k(x_i)\right)$ 3.5-10

- $H_L(t|x_i) = \int_0^t P_L\left(\tau|k(x_i)\right)\ \lambda(t-\tau\,|\,x_i) \begin{cases} r(t-\tau\,|\,x) & : x_i \text{ non-transient fault} \\ 1. & : x_i \text{ transient fault} \end{cases} d\tau$ 3.5-11

$= a_L(t|x_i) M_L^{o}\left(t|k(x_i)\right) + b_L(t|x_i) M_L^{1}\left(t|k(x_i)\right) + c_L(t|x_i) M_L^{2}\left(t|k(x_i)\right)$ 3.5-12

- $h_F(t|x_i) = \int_0^t P_F\left(\tau|k(x_i)\right)\, \lambda(t-\tau|x_i) \begin{cases} r(t-\tau|x) & : x_i \text{ non-transient fault} \\ 1. & : x_i \text{ transient fault} \end{cases} d\tau$ 3.5-13

$= a_F(t|x_i) m_F^0\left(t|k(x_i)\right) + b_F(t|x_i) m_F^1\left(t|k(x_i)\right) + c_F(t|x_i) m_F^2\left(t|k(x_i)\right)$ 3.5-14

- $g_F(t|x_i) = \frac{h_F(t|x_i)}{a(t|x_i)} \begin{cases} \frac{1.}{1.-r(t|x)} & : x_i \text{ non-transient fault} \\ 1. & : x_i \text{ transient fault} \end{cases}$ 3.5-15

- $H_{\bar{B}}(t|x_i) = \int_0^t P_{\bar{B}}\left(\tau \mid k(x_i)\right)\, \lambda(t-\tau|x_i) \begin{cases} r(t-\tau|x) & : x_i \text{ non-transient fault} \\ 1. & : x_i \text{ transient fault} \end{cases} d\tau$ 3.5-16

$= a_{\bar{B}}(t|x_i) M_{\bar{B}}^0\left(t|k(x_i)\right) + b_{\bar{B}}(t|x_i) M_{\bar{B}}^1\left(t|k(x_i)\right) + c_{\bar{B}}(t|x_i) M_{\bar{B}}^2\left(t|k(x_i)\right)$ 3.5-17

$$\bullet \quad \overline{G_B}(t|x_i) = \frac{\overline{H_B}(t|x_i)}{a(t|x_i)} \begin{cases} \dfrac{1.}{1.-r(t|x)} & : \ x_i \text{ non-transient fault} \\ \\ 1. & : \ x_i \text{ transient fault} \end{cases} \qquad 3.5\text{-}18$$

$$\bullet \quad H_B(t|x_i) = \int_0^t P_B\left(\tau \,|\, k(x_i)\right) \lambda\,(t-\tau\,|\,x_i) \begin{cases} r(t-\tau|x) & : \ x_i \text{ non-transient fault} \\ \\ 1. & : \ x_i \text{ transient fault} \end{cases} d\tau \qquad 3.5\text{-}19$$

$$= a_B(t|x_i) M_B^0\left(t|k(x_i)\right) + b_B(t|x_i) M_B^1\left(t|k(x_i)\right) + c_B(t|x_i) M_B^2\left(t|k(x_i)\right) \qquad 3.5\text{-}20$$

$$\bullet\ h_{DF}(t|x_i,y_j) = \int_0^t P_{DF}(\tau|k(x_i),k(y_j))H_B(t-\tau|x_i)\lambda(t-\tau|y_j) \begin{cases} r(t-\tau|x)r(t-\tau|y) & : x_i \text{ and } y_j \text{ are non-transient faults} \\ r(t-\tau|x) & : x_i \text{ non-transient and } y_j \text{ transient faults} \\ r(t-\tau|y) & : x_i \text{ transient and } y_j \text{ non-transient faults} \\ 1. & : x_i \text{ and } y_j \text{ are transient faults} \end{cases} d\tau \qquad 3.5\text{-}21$$

$$= a_{DF}(t|x_i,y_j)m^0_{DF}\left(t|k(x_i),k(y_j)\right) + b_{DF}(t|x_i,y_j)m^1_{DF}\left(t|k(x_i)(k(y_j)\right) \qquad 3.5\text{-}22$$
$$+ c_{DF}(t|x_i,y_j)m^2_{DF}\left(t|k(x_i)k(y_j)\right)$$

$$\bullet\ K(t|\underline{\ell}) = \sum_y \sum_j C\left(t|\underline{\ell}-\underline{1}(y),y_j\right) P^*\left(t|\underline{\ell}-\underline{1}(y)\right)\left(n(y)-\ell(y)+1\right)\lambda(t|y_j) \qquad 3.5\text{-}23$$
$$+ \left[A'(t|\underline{\ell}) + a'(t|\underline{\ell})\right] P^*(t|\underline{\ell})$$

The evaluation of $K(t|\underline{\ell})$ requires the calculation of $P^*(t|\underline{\ell})$ and $P^*\left(t|\underline{\ell}-1(y)\right)$ for y=1,...,N; in subroutine SUMAT, N calls are made to subroutine FPSTAR to compute $P^*\left(t|\underline{\ell}-\underline{1}(y)\right)$ and one call is made to subroutine FPSTREC to compute $P^*(t|\underline{\ell})$ from $P^*\left(t|\underline{\ell}-\underline{1}(N)\right)$ by the recurrence relation:

$$P^*(t|\underline{\ell}) = \frac{n(y)-\ell(y)+1}{\ell(y)} \frac{(1.-r(t|y))}{r(t|y)} P^*\left(t|\underline{\ell}-\underline{1}(y)\right) \qquad 3.5\text{-}24$$

This approach requires N calculations of the P* function, which as equation 3.5-7 shows requires the evaluation of a combinatorial term. An improved procedure would be to compute $P^*(t|\underline{\ell})$ once using subroutine FPSTAR and then compute the $P^*\left(t|\underline{\ell}-\underline{1}(y)\right)$ by equation 3.5-24; this approach requires only one calculation of a combinatorial term.

- $$C(t|\underline{\ell},y_j) = \sum_x \sum_i G_B^-(t|x_i) D(t|\underline{\ell},x_i,y_j) \qquad 3.5\text{-}25$$

$$= \sum_x \sum_i \frac{H_B^-(t|x_i)}{a(t|x_i)} D(t|\underline{\ell},x_i,y_j) \begin{cases} \dfrac{1.}{1.-r(t|x)} & : x_i \text{ non-transient fault} \\ 1. & : x_i \text{ transient fault} \end{cases} \qquad 3.5\text{-}26$$

The CARE-III documents give a different definition for $C(t|\underline{\ell},y_j)$, i.e.,

- $$C(t|\underline{\ell},y_j) = \sum_x \sum_i \frac{H_B(t|x_i)}{H_L(t|x_i)} D(t|\underline{\ell},x_i,y_j), \qquad 3.5\text{-}27$$

BCS is currently investigating this apparent problem.

- $$A'(t|\underline{\ell}) = \sum_x \sum_i \sum_y \sum_j \frac{h_{DF}(t|x_i,y_j)}{H_L(t|x_i)H_L(t|y_j)} B(t|\underline{\ell},x_i,y_j) \qquad 3.5\text{-}28$$

- $$a'(t|\underline{\ell}) = \sum_x \sum_i a(t|x_i)g_F(t|x_i) \cdot \begin{cases} \ell(x) & : x_i \text{ non-transient fault} \\ n(x)-\ell(x) & : x_i \text{ transient fault} \end{cases} \qquad 3.5\text{-}29$$

$$= \sum_{x} \sum_{i} h_F(t|x_i) \cdot \begin{cases} \dfrac{\ell(x)}{1.-r(t|x)} & : x_i \text{ non-transient fault} \\ \\ n(x)-\ell(x) & : x_i \text{ transient fault} \end{cases} \qquad 3.5\text{-}30$$

BCS has identified a programming error in subroutine FAPC which computes $a'(t|\underline{\ell})$; this error has been corrected on the BCS version of CARE-III and NASA has been informed of the problem.

- $$D(t|\underline{\ell},x_i,y_j) = \sum_{\substack{\mu(x) \\ \mu(y)}} b_{xy}\left(\ell(x)-\mu(x),\ell(y)-\mu(y)\right) P\left(\mu(x),t|\ell(x)\right) P\left(\mu(y),t|\ell(y)\right) \begin{cases} \mu(x)\dfrac{a(t|x_i)}{a(t|x)} & : \text{non-transient} \\ \\ \left(n(x)-\ell(x)\right) a(t|x_i) & : \text{transient} \end{cases}$$

  3.5-31

- $$B(t|\underline{\ell},x_i,y_j) = \sum_{\substack{\mu(x) \\ \mu(y)}} b_{xy}\left(\ell(x)-\mu(x),\ell(y)-\mu(y)\right) P\left(\mu(x),t|\ell(x)\right) P\left(\mu(y),t|\ell(y)\right) C(x_i,y_j) a(t|x_i) a(t|y_j)$$

  3.5-32

where

$$C(x_i,y_j) = \begin{cases} \dfrac{\mu(x)\,\mu(y)}{a(t|x)a(t|y)} & : x \neq y,\ x_i \text{ and } y_j \text{ non-transient faults} \\ \dfrac{\mu(x)\,(\mu(x)-1.)}{a(t|x)a(t|x)} & : x = y,\ x_i \text{ and } y_j \text{ non-transient faults} \\ (n(x)-\ell(x))(\mu(y)-\ell(y)) & : x \neq y,\ x_i \text{ and } y_j \text{ transient faults} \\ (n(x)-\ell(x))\,(n(x)-\ell(x)-1) & : x = y,\ x_i \text{ and } y_j \text{ transient faults} \\ \dfrac{\mu(x)\,(n(y)-\ell(y))}{a(t|x)} & : x_i \text{ non-transient and } y_j \text{ transient faults} \\ \dfrac{(n(x)-\ell(x))\,\mu(y)}{a(t|y)} & : x_i \text{ transient and } y_j \text{ non-transient faults} \end{cases} \qquad 3.5\text{-}33$$

3.6 COMPUTATIONAL METHODS

In the previous section the implementation of the reliability model in the CARE3 program was outlined; now, the computational methods used to solve the model equations will be reviewed. The objective of this section is to document the numerical procedures that are implemented and present the BCS evaluation of these algorithms in the CARE-III environment. In the following sections the numerical procedures are first highlighted and then discussed in detail.

3.6.1 Overview of Algorithms

In this section each of the numerical procedures used in the CARE3 program is highlighted. The requirements for the algorithm and its implementation are discussed first, followed by a preview of the BCS analysis of the algorithm in the CARE-III context. In the succeeding sections a detailed description and analysis of each algorithm is provided.

- Numerical Integration

  The calculation of the unreliability $Q(t \mid \underline{\ell})$ for a fault vector $\underline{\ell}$ requires the calculation of the integral of $K(t \mid \underline{\ell})$:

$$Q(t \mid \underline{\ell}) = \int_0^t K(\tau \mid \underline{\ell})\, d\tau \qquad (3.6\text{-}1)$$

  where $K(t \mid \underline{\ell})$ can be computed from the reliability data, critical pairs fault tree(s) data and the output of the coverage model (see Section 3.2). The numerical integration procedure is based on Simpson's Rule and is implemented in subroutines UNRELQ, SUMMAT and FINTGRT. Both $Q(t \mid \underline{\ell})$ and $K(t \mid \underline{\ell})$ are computed at the (equally spaced with stepsize Δt) discrete time points for the reliability model.

BCS has carefully reviewed UNRELQ, FINTGRT and SUMMAT and the subroutines which they use; no programming errors in the implementation of Simpson's Rule were detected by the review. However, a programming error in subroutine FAPC (called by SUMMAT) was discovered; this error is discussed in Section 3.5.5.

- Numerical Convolution

The calculation of $K(\underline{\ell}\,|t)$ for a fault vector $\underline{\ell}$ requires several calculations of the convolution of two functions, $P_1(t)$ and $P_2(t)$:

$$y(t) = \int_0^t P_2(\tau)\, P_1(t - \tau)\, d\tau \qquad (3.6\text{-}2)$$

where $P_1(t)$ is the measure of the rate at which a certain class of fault occurs and $P_2(\tau)$ is a function of the interval, τ, between that occurrence and the entry of the fault into a particular coverage-model state. Each of the output coverage functions, P_{Dp}, P_L, P_F, P_B^-, P_B and P_{DF}, enter into such a convolution calculation (see Section 3.2). The numerical convolution procedure is based on the method of moments and is implemented in subroutines, FHSFST, FHDFST, ABCST, FFSFST and FFDFST.

BCS has carefully reviewed FHSFST, FHDFST, ABCST, FFSFST and FFDFST and the subroutines which they use; no programming errors in the implementation of the method of moments were detected by the review. However, BCS has raised questions about the accuracy of the method of moments for calculating the required convolutions; these questions are addressed in detail in Section 3.6.3.

3.6.2 Numerical Integration

The numerical integration procedure used in the CARE3 program is based on Simpson's Rule and is implemented in subroutines UNRELQ, SUMMAT and FINTGRT, specifically to compute the integrals:

$$Q(t_j \mid \underline{\ell}) = \int_0^{t_j} K(\tau \mid \underline{\ell}) d\tau \quad j = 1, 2, \ldots, j_{max}, \tag{3.6-3}$$

where $\underline{\ell}$ is a fault vector and t_j; $j = 1, 2, \ldots, j_{max}$ are equally spaced discrete time points for the reliability model:

$$t_j = (j - 1)\Delta t \; ; \quad j = 1, 2, \ldots, j_{max} \tag{3.6-4}$$

The function $K(t \mid \underline{\ell})$ is evaluated by subroutine SUMMAT (see Section 3.2 for a description of the calculations involved) and the integrals are evaluated as follows by subroutine UNRELQ:

- Case 1: $j = 1$

$$Q(t_1 \mid \underline{\ell}) = 0. \quad , \tag{3.6-5}$$

- Case 2 = $j > 1$, j even

$$Q(t_j \mid \underline{\ell}) = \sum_{\substack{k=2 \\ k \text{ even}}}^{j} F_k \tag{3.6-6}$$

- Case 3 = $j > 1$, j odd

$$Q(t_j \mid \underline{\ell}) = \sum_{\substack{k=3 \\ k \text{ odd}}}^{j} F_k \tag{3.6-7}$$

where the F_k are computed in subroutine FINTGRT:

- Case 1 : $k = 1$

$$F_k = 0. \tag{3.6-8}$$

- Case 2 : k = 2

$$F_k = \frac{\Delta t}{2} \left(K(t_1 \mid \underline{\ell}) + K(t_2 \mid \underline{\ell})\right) \tag{3.6-9}$$

- Case 3 : k > 2

$$F_k = \frac{\Delta t}{3} \left(K(t_{k-2} \mid \underline{\ell}) + K(t_{k-1} \mid \underline{\ell}) + K(t_k \mid \underline{\ell})\right) \tag{3.6-10}$$

3.6.3 Numerical Convolution

The numerical convolution procedure used in the CARE3 program is based on the method of moments and is implemented in subroutines FHSFST, FHDFST, ABCST, FFSFST and FFDFST specifically to compute the convolutions:

$$y(t_j) = \int_0^{t_j} P_2(\tau)\, P_1(t - \tau)\, d\tau, \tag{3.6-11}$$

where $P_1(t)$ is a reliability model function of the form:

$$P_1(t) = \begin{cases} \lambda_{x_i}(t) & : \text{single fault,} \\ \lambda_{x_i y_j}(t) & : \text{double fault,} \end{cases} \tag{3.6-12}$$

$P_2(t)$ is one of the coverage model output functions, P_{DP}, P_L, P_F, P_B, $P_{\bar{B}}$ and P_{DF}, and t_j; $j = 1, 2, \ldots, j_{max}$ are equally spaced discrete time points for the reliability model:

$$t_j = (j - 1)\,\Delta t \;;\quad j = 1,2,\ldots., j_{max}. \tag{3.6-13}$$

The procedure is based on the critical assumptions that $P_1(t)$ is a much more slowly varying function of time than is $P_2(t)$ and that $P_2(t)$ decays rapidly to zero:

$$P_2(\tau) \approx 0. \qquad : \tau > t_0 > 0., \tag{3.6-14}$$

$$P_1(t-\tau) \approx a(t) + \tau b(t) + \tau^2 c(t) \quad : 0 \leq \tau \leq t_0,\ t_0 > 0. \tag{3.6-15}$$

Under these assumptions:

$$y(t_j) = \int_0^{t_0} P_2(\tau)\ (a(t_j) + \tau b(t_j) + \tau^2 c(t_j))d\tau, \tag{3.6-15}$$

$$= a(t_j)\int_0^{t_0} P_2(\tau)d\tau + b(t_j)\int_0^{t_0} \tau\ P_2(\tau)d\tau + c(t_j)\int_0^{t_0} \tau^2 P_2(\tau)d\tau, \tag{3.6-16}$$

$$= a(t_j)\ M_2^0(t_0) + b(t_j)\ M_2^1(t_0) + c(t_j)\ M_2^2(t_0), \tag{3.6-17}$$

where

$$M_2^i(t_0) = \int_0^{t_0} \tau^i\ P_2(\tau)d\tau\ ; \quad i = 0,1,2. \tag{3.6-18}$$

The time t_0 and the coefficients, $a(t_j)$, $b(t_j)$ and $c(t_j)$ are computed by subroutine ABCST to make the approximation to $P_1(t_j - \tau)$ exact at $t_j - t_0$, $t_j - (t_0/2)$ and t_j. Subroutines FFSFST and FFDFST are called by ABCST to compute $P_1(t)$ for the single or double fault cases, respectively. Finally, the approximate values of $y(t_j)$ are evaluated by subroutines FHSFST and FHDFST for the single or double fault cases, respectively.

BCS has raised questions about the accuracy of the method of moments, as implemented in CARE-III, for computing the convolution in equation 3.6-11. Since this convolution provides the crucial links between the coverage model and the reliability model, its accurate evaluation is important for

the CARE-III estimate of reliability. The assumption that $P_1(t)$ is a much more slowly varying function of time than is $P_2(t)$ is consistent with the CARE-III assumption that coverage rates are much higher than module failure rates; therefore, this assumption should not degrade the accuracy of the solution.

However, the assumption that $P_2(\tau)$ decays rapidly to zero may not be valid for all coverage model parameters; indeed for some cases $P_2(\tau)$ decays to a non-zero steady state value. In such a case two sources of error develop in the calculation; first, the contribution to the convolution for $\tau > t_o$, which is neglected, may become significant and second, the approximation to $P_1(t)$ becomes less accurate as t_o becomes larger. In the CARE-III implementation this potential source of error is not estimated or monitored.

BCS is currently investigating the impact of this problem on the CARE-III reliability estimate by running cases with coverage models that do not satisfy the assumptions in equations 3.6-14 to 3.6-15.

Section 4
COVERAGE MODEL

The theory and implementation of the CARE-III coverage model is described in this section. The mathematical details of the coverage model have been extracted from the CARE-III documentation, J. J. Stiffler, L. A. Bryant and L. Guccione (1979), J. J. Stiffler and L. A. Bryant (1982), J. J. Stiffler, J. S. Neumann and L. A. Bryant (1982) and the CARE-III program (Version 3, 1982). In those areas where the documentation is vague or incomplete, BCS has completed the model specifications based on its understanding of the applicable reliability methods.

Sections 4.1 and 4.2 present the single and double fault coverage models and their mathematical solution in terms of a sequence of Volterra Integral Equations of the second kind. The implementation of the coverage models is described in Sections 4.3 and 4.4, first an overiew of the COVRGE program is presented in Section 4.3 and then the computational methods used in COVRGE are highlighted in Section 4.4.

Any discrepencies between the CARE-III documentation or the COVRGE program and the coverage model, as found by BCS during the Task 1 review, are pointed out in the discussion in Section 4.

4.1 STOCHASTIC COVERAGE MODEL

The two Coverage Models are used to analyze the latency period of a fault and the interaction of latent faults in a critical pair. These models are described in detail in Sections 3.1.3 and 3.1.4, and are used to derive the transition rates for the Aggregate Model as shown in Section 3.4.

The coverage functions needed in the calculation of these rates are the state probabilities for different coverage states and the intensities of entry into failure state.

All the calculations are done within the context of the coverage dynamics, independently of the dynamics of the rest of the system, so the reversibility argument given in Section 3.1.6 for transient faults (backward transition in vector $\underline{\ell}$) is irrelevant for the Coverage Models. The functions to be obtained in Section 4.2 are then valid for both transient and non-transient faults.

More specifically the functions needed from each of the Coverage Models are as follows:

- From the Single Fault Coverage Model, SFCM, and for each fault category x_i:

  The intensity of entry at time t into failure state F: $p_F(t)$; and the probabilities that at time t, the fault is in
  benign state B: $P_B(t)$,
  non-benign state $\overline{B}$: $P_{\overline{B}}(t)$,
  latent state L: $P_L(t)$,
  detected as permanent state DP: $P_{DP}(t)$.

Where $\overline{B}$ = A or E,

and $L = \begin{cases} B \text{ or } \overline{B} & \text{for a non-transient fault,} \\ \overline{B} & \text{for a transient fault.} \end{cases}$

- From the Double Fault Coverage Model, DFCM, and for each pair of fault categories x_i, y_j:

  The intensity of entry at time t into the Double-Fault failure state DF: $p_{DF}(t)$.

In all the above functions, time t is measured from time of first entry into state A for SFCM, and first entry into state B_1 A_2 for DFCM.

Holding times in each state, though not necessarily exponentially distributed, are independent from past dynamics and so the Coverage Models, SFCM and DFCM, are homogeneous semi-Markov processes with respective initial States A and B_1A_2. In Section 4.2 it is shown how properties of semi-Markov processes - see Appendix - are used to derive the functions that are of interest to solve the reliability problem.

4.2 SOLUTION OF COVERAGE MODELS

4.2.1 Single Fault Coverage Model

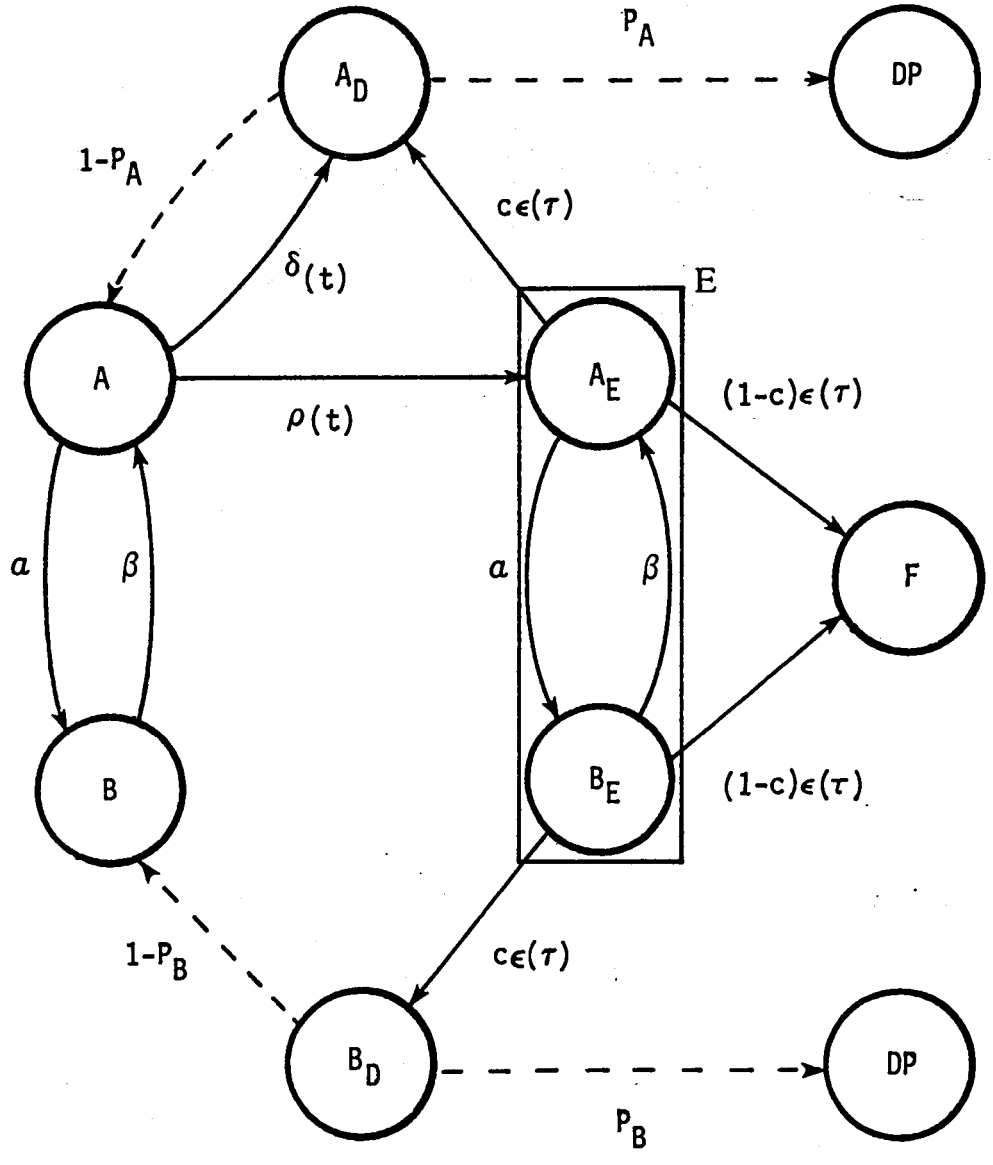
The Single Fault Coverage Model is shown in Figure 4.2-1. Transitions out of each state are measured from the time of entry into it, except for the two error states. In this case transitions are measured from time of entry into the active-error state A_E. The essence of this interpretation is that detection schedules are independent of fluctuations between active and benign states. The model can be reduced by replacing the two error states by one E shown in Figure 4.2-1 as a rectangular block. This interpretation although not consistent with the description of the SFCM given by J. J. Stiffler and L. A. Bryant (1982), coincides with the present version of CARE III.

The input parameters (α, β, $\delta(t)$, $\rho(t)$, $\epsilon(t)$, C, P_A, P_B) determine the transition probability distributions $Q_{ij}(t)$. The derivatives of $Q_{ij}(t)$ and the holding time distributions $h_i(t)$ are given in Table 4.2-1.

These are then used to calculate the first entry or return distributions $F_{ij}(t)$ or the entry intensities $f_{ij}(t)$, and from these the state probabilities $P_{ij}(t)$.

The model corresponding to the case $P_A = P_B = 1$ is simpler and affords an easier solution. The general model can then be solved from the simpler one as follows.

Let $F_x(t)$ denote the probability of being in state x at time t, given that $P_A = P_B = 1$ (i.e., given that the faulty element has not yet been returned to active state A after possible detection); $G_x(t)$ denote the probability of the same event but under no restrictions on P_A and P_B; and A(t) denote the intensity of first return to active state A after detection.



A: ACTIVE
B: BENIGN
D: DETECTED
E: ERROR
F: FAILURE
DP: DETECTED AS PERMANENT (NON-TRANSIENT)

t = time from entry into active state A

τ = time from entry into error state E

Figure 4.2-1 Single Fault Coverage Model

TABLE 4.2-1 Single-Fault Coverage Model
Transition probability densities $Q'_{ij}(t)$
and holding time distributions $h_i(t)$

From State i \ To State j	1.A	2.B	3.E	4.A_D	5.B_D	6.F	7.DP	$h_i(t)$
1. A	-	$\alpha\bar{e}^{\alpha t}d(t)r(t)$	$\bar{e}^{\alpha t}d(t)\rho(t)$	$\bar{e}^{\alpha t}\delta(t)r(t)$	-	-	-	$\bar{e}^{\alpha t}d(t)r(t)$
2. B	$\beta\bar{e}^{\beta t}$	-	-	-	-	-	-	$\bar{e}^{\beta t}$
3. E	-	-	-	$C\epsilon(t)f(t)$	$C\epsilon(t)(1-f(t))$	$(1-C)\epsilon(t)$	-	$e(t)$
4. A_D	$(1-P_A)\Delta(t)$	-	-	-	-	-	$P_A\Delta(t)$	1
5. B_D	-	$(1-P_B)\Delta(t)$	-	-	-	-	$P_B\Delta(t)$	1
6. F	-	-	-	-	-	-	-	1
7. DP	-	-	-	-	-	-	-	1

$$f(t) = \frac{\beta + \alpha \exp(-(\alpha+\beta)t)}{\alpha + \beta}$$

$\Delta(t)$ = Dirac's delta

Then it follows that for any state x different from B,

$$G_x(t) = F_x(t) + \int_o^t A(u)\, G_x(t-u)\, du$$

where the integral accounts for all possible cycles through the system until ending in state x at time t.

It also follows that

$$A(t) = (1-P_A)\, \psi_A(t) + (1-P_B)\int_o^t \psi_B(u)\ \beta \exp\left[-\beta(t-u)\right] du,$$

where $\psi_A(t)$ and $\psi_B(t)$ are the intensities of entry into states A_D and B_D respectively, given $P_A = P_B = 1$.

For the case of state B a modification should be made. Instead of using $F_B(t)$, use $F_B(t) + \chi_B(t)$,

where $\chi_B(t)$ = Probability of entering B_D for the first time and then remaining in the benign state until time t

$$= (1-P_B)\int_o^t \psi_B(u) \exp\left[-\beta(t-u)\right] du.$$

In Table 4.2-2, a summary of the Single-Fault Model Equations is given with the corresponding definitions and mathematical expressions.

As an example the first two formulas are derived.

$\phi(t)$ is β^{-1} times the intensity of reentry into state A exactly t time units after previous entry.

and $P_a(t)$ is the probability of being in state A at time t, when $P_A = P_B = 1$.

Using the notation given in the Appendix and the numbering of states in Table 4.2-1, these terms are $\beta^{-1} f_{11}(t)$ and $P_{11}(t)$ respectively. So for the first function,

$$\beta\phi(t) = f_{11}(t) = \int_0^t Q_{12}(dx)\ f_{21}(t-x),$$

where

$$Q'_{12}(t) = \alpha \exp(-\alpha t)\ d(t)\ r(t),$$

and $f_{21}(t) = Q'_{21}(t) = \beta \exp(-\beta t).$

Hence

$$\phi(t) = \alpha \exp(-\beta t) \int_0^t \exp\left[-x(\alpha - \beta)\right] d(x)\ r(x)\ dx.$$

Similarly for the second function,

$$\begin{aligned} P_a(t) = P_{11}(t) &= h_1(t) + \int_0^t F_{11}(dx)\ P_{11}(t-x) \\ &= h_1(t) + \int_0^t f_{11}(x)\ P_{11}(t-x)\ dx \\ &= \exp(-\alpha t)\ d(t)\ r(t) + \int_0^t \phi(t-x)\ P_a(x)\ dx. \end{aligned}$$

The mathematical expressions shown in Table 4.2-2 for the functions $\chi_B(t)$ and $F_\chi(t)$ differ from those given by J. J. Stiffler and L. A. Bryant (1981), and from those implemented in the present version of CARE-III. The suggested changes account for possible returns to the benign state B from state B_D with probability $1-P_B$ and thus affect the calculation of the probability of being in the benign state at time t: $P_B(t)$. The present implemented version is only valid if $P_B = 0$.

Table 4.2-2
Single-Fault Model Equations

FUNCTION	MATHEMATICAL EXPRESSION*	DEFINITION
$\phi(t)$	$\alpha e^{-\beta t} \int_0^t e^{-(\alpha-\beta)u} r(u)\, d(u)\, du$	β^{-1} Times the Probability intensity of re-entering state A exactly t time units after the previous entry.
$P_a(t)$	$e^{-\alpha t} r(t)\, d(t) + \beta \int_0^t \phi(t-u)\, P_a(u)\, du$	Probability of being in state A at time t when $P_A = P_B = 1$
$P_b(t)$	$\phi(t) + \beta \int_0^t \phi(t-u)\, P_b(u)\, du$	Probability of being in state B at time t when $P_A = P_B = 1$
$P_e(t)$	$\int_0^t e^{-\alpha u} \rho(u)\, d(u)\, e(t-u)\, du + \beta \int_0^t \phi(t-u)\, P_e(u)\, du$	Probability of being in state A_E or B_E at time t when $P_A = P_B = 1$

* t Here is a measure of the time since the entry into state A.

Table 4.2-2 (Continued)
Single-Fault Model Equations

FUNCTION	MATHEMATICAL EXPRESSION*	DEFINITION
$p_e(t)$	$e^{-\alpha t}\,\rho(t)\,d(t) + \beta \int_0^t \phi(t-u)\,p_e(t)\,du$	Intensity of entry into state A_E at time t when $P_A = P_B = 1$
$p_e^-(t)$	$e^{-\alpha t}\,\delta(t)\,r(t) + \beta \int_0^t \phi(t-u)\,p_e^-(u)\,du$	Intensity of entry into state A_D from state A at time t when $P_A = P_B = 1$
$P_f(t)$	$(1-C) \int_0^t p_e(u)\,\epsilon(t-u)\,du$	Intensity of entry into state F at time t when $P_A = P_B = 1$
$\Psi_A(t)$	$C \int_0^t p_e(u)\,\epsilon(t-u)\,\frac{\beta+\alpha e^{-(\alpha+\beta)(t-u)}}{\alpha+\beta}\,du + p_e^-(t)$	Intensity of entry into state A_D at time t for the first time

* t Here is a measure of the time since the entry into state A.

Table 4.2-2 (Continued)
Single-Fault Model Equations

FUNCTION	MATHEMATICAL EXPRESSION*	DEFINITION
$\psi_B(t)$	$\frac{\alpha C}{\alpha+\beta} \int_0^t P_e(u)\ (1 - e^{-(\alpha+\beta)\ (t-u)})\ \epsilon\ (t-u)\ du$	Intensity of entry into state B_D at time t for the first time
$\chi_B(t)$	$(1-P_B) \int_0^t \psi_B(u) e^{-\beta(t-u)} du$	Probability of having entered state B_D for the first time and then remaining in the benign state until time t
$P_{dp}(t)$	$P_A \int_0^t \psi_A(u)du + P_B \int_0^t \psi_B(u)du$	Probability that a fault has been diagnosed as permanent by time t
$F_X(t)$	$F_x(t) + \int_0^t \left[(1-P_A)\psi_A(t-u) + \beta \chi_B\ (t-u)\ F_X(u)du\right.$	Function relating probabilities and intensities derived when $P_A = P_B = 1$ to those same quantities when P_A & P_B are arbitrary

* t Here is a measure of the time since the entry into state A.

Table 4.2-2 (Continued)
Single-Fault Model Equations

FUNCTION	MATHEMATICAL EXPRESSION*	DEFINITION
$P_B(t)$	$F_X(t)$ with $F_x(t) = P_b(t) + \chi_B(t)$	Probability of being in state B at time t
$P_{\bar{B}}(t)$	$F_X(t)$ with $F_x(t) = P_a(t) + P_e(t)$	Probability of being in a non-benign state at time t
$P_L(t)$	$F_X(t)$ with $F_x(t) =$ $P_b(t) + \chi_B(t)$ + $P_a(t) + P_e(t)$ NON TRANSIENT FAULTS $P_a(t) + P_e(t)$ TRANSIENT FAULTS	Probability of a latent fault or undetected error at time t
$P_{DP}(t)$	$F_X(t)$ with $F_x(t) = P_{dp}(t)$	Probability that a fault has been diagnosed as permanent by time t

* t Here is a measure of the time since the entry into state A.

4.2.2 Double-Fault Coverage Model

A detailed version of the DFCM, consistent with the SFCM, is shown in Figure 4.2-2. CARE-III considers a simplified version whereby a detected as non-permanent fault causes immediate failure of the system, e.g., from state $B_1\ D_2$ there is an instantaneous transition either to state $B_1\ DP_2$ with probability P_{A_2} or to state DF with probability $1\text{-}P_{A_2}$. This change is represented by the dashed lines into state DF. This new model will result on a higher intensity of entry into state DF and hence a smaller (conservative) value for the Reliability of the system.

The holding times for this model follow the stochastic characteristics of those in the SFCM and so the Double-Fault Model is also a semi-Markov process. The transition probability densities, $Q'_{ij}(t)$, are given in Table 4.2-3.

Figure 4.2-3 represents the Double Fault Coverage Model as given by J. J. Stiffler and L. A. Bryant (1982). It should be noted that in this representation, the parameters for transitions into states F and D do not correspond to competing transitions as is the case for such graphical representations of semi-Markov processes.

Using the formulas of the Appendix, and given the functions $c_1(t)$, $c_3(t)$, $c_4(t)$ and $f_1(t)$ as defined in Table 4.2-4, it follows that

$$p_{DF}(t) = c_1(t) + \int_0^t f_1(x)\ p_4(t\text{-}x)\ dx,$$

where $p_4(t)$, the intensity of entry into state F t units of time after entry into state $B_1\ B_2$, is given by

$$p_4(t) = c_4(t) + \int_0^t c_3(x)\ p_4(t\text{-}x)\ dx.$$

Straightforward analysis, e.g., using Laplace transforms, show that these formulas are equivalent to those given in Table 4.2-4.

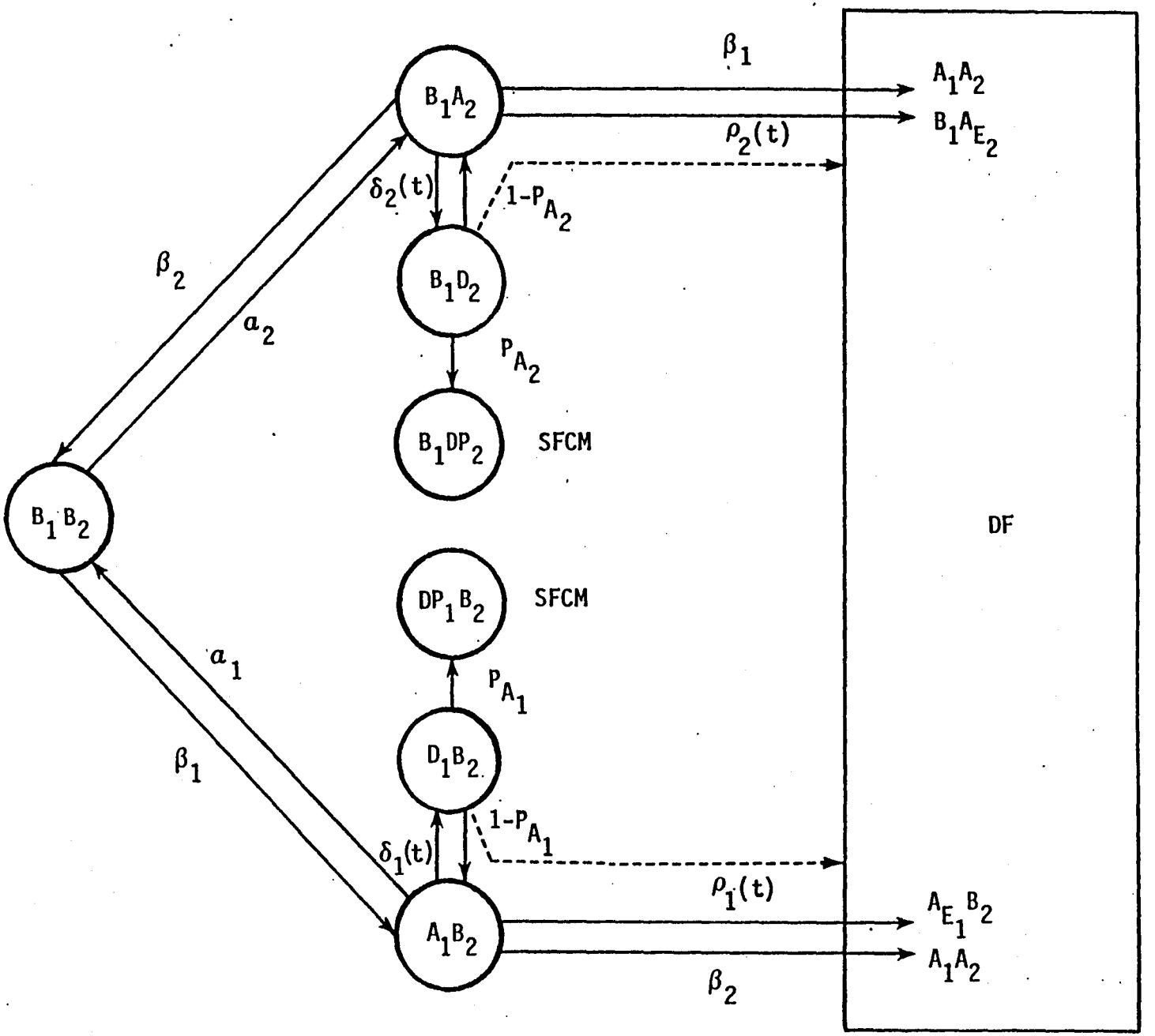


Figure 4.2-2 Double Fault Coverage Model

TABLE 4.2-3 Double Fault Coverage Model

Transition Probability Densities $Q_{ij}(t)$

From \ to	B_1A_2	A_1B_2	B_1B_2	D	F
B_1A_2	-	-	$f_1(t)$	$P_{A_2}\,\delta_2(t)a_2(t)r_2(t)b_1(t)$	$c_1(t)$
A_1B_2	-	-	$f_2(t)$	$P_{A_1}\,\delta_1(t)a_1(t)r_1(t)b_2(t)$	$c_2(t)$
B_1B_2	$\beta_2(t)\,b_1(t)$	$\beta_1(t)\,b_2(t)$	-	-	-
D	-	-	-	-	-
F	-	-	-	-	-

$c_i(t)$, $f_i(t)$ as given in table 4.2-4

$\beta_i(t) = \beta_i \exp(-\beta_i t)$ $\qquad b_i(t) = \exp(-\beta_i t)$

$\alpha_i(t) = \alpha_i \exp(-\alpha_i t)$ $\qquad a_i(t) = \exp(-\alpha_i t)$

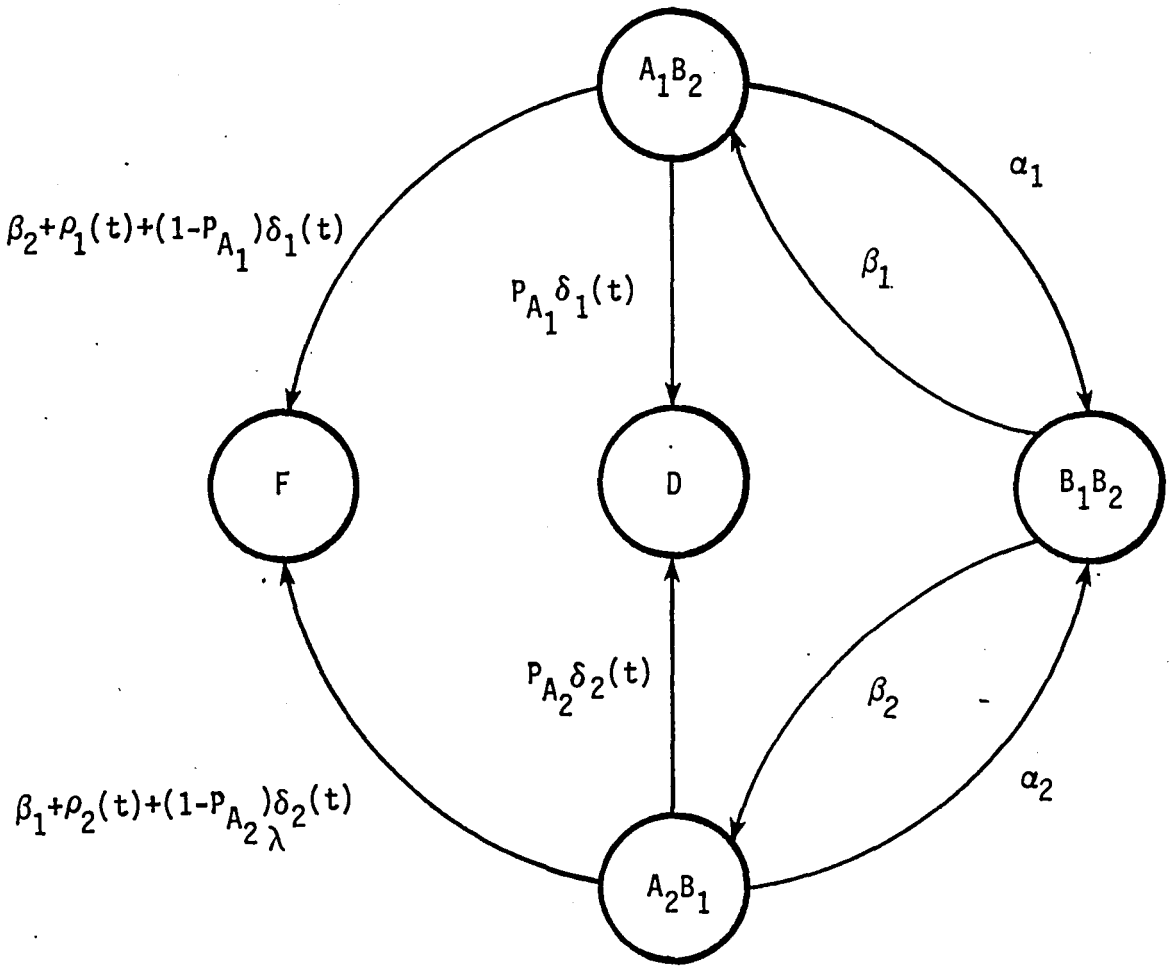


Figure 4.2-3 CARE III Double Fault Coverage Model

Table 4.2-4

Double-Fault Model Equations

FUNCTION	MATHEMATICAL EXPRESSION*	DEFINITION
$c_i(t)$	$\beta_i(t)d_j(t)r_j(t)a_j(t)$ +	Transition density from state A_jB_i to state F
$i = 1,2$	$(1-P_{A_j})b_i(t)\,\delta_j(t)r_j(t)a_j(t)$ +	
$j = 3-i$	$b_i(t)d_j(t)\,\rho_j(t)a_j(t)$	
$f_i(t)$	$\alpha_j(t)b_i(t)d_j(t)r_j(t)$	Transition density from state A_jB_i to state B_1B_2
$i = 1,2$		
$j = 3-i$		
$c_4(t)$	$\int_0^t \left[c_1(t-u)\,\beta_2(u)b_1(u) + c_2(t-u)\,\beta_1(u)b_2(u)\right] du$	Intensity of entry into state F t time units after last entry into state B_1B_2

Table 4.2-4 (Continued)
Double-Fault Model Equations

FUNCTION	MATHEMATICAL EXPRESSION*	DEFINITION
$c_3(t)$	$\int_0^t \left[f_1(t-u)\,\beta_2(u)b_1(u) + f_2(t-u)\,\beta_1(u)b_2(u) \right] du$	Intensity of re-entry into state B_1B_2 t time units after a previous entry
$P_3(t)$	$f_1(t) + \int_0^t c_3(t-u)p_3(u)du$	Intensity of last entry into state B_1B_2 t time units after entry into state A_2B_1
$P_{DF}(t)$	$c_1(t) + \int_0^t c_4(t-u)p_3(u)du$	Intensity of entry into state F t time units after entry into state A_2B_1

4.3 IMPLEMENTATION IN CARE-III

In the previous sections the formulation of the coverage model was reviewed; now, the implementation of the model in the program, COVRGE, will be described in some detail. The objective of this section is to document the model that is actually implemented and outline the calculations performed. In the following sections, the overall structure and data flow of the COVRGE program are described, the solution of the coverage model is outlined and the basic coverage functions are defined.

4.3.1 Overview of COVRGE Program

Figure 4.3-1 illustrates the overall data flow for the COVRGE program. The user's input data for the coverage model is read from file CREIN by the input program CAREIN; it includes the fault type parameters: α, β, δ, ρ, ϵ, δ_F, ρ_F and ϵ_F. After the data is checked and preprocessed by CAREIN, it is passed to COVRGE on file COVIN. Then the coverage model is solved for each fault type by COVRGE and the functions p_{DP}, P_L, p_F, P_B, $P_{\overline{B}}$ and p_{DF} are computed. The moments of the coverage functions are computed and passed to the reliability program CARE3, on file CVGMTS. In addition the coverage functions are passed to the plotting program, CVGPLT, on files SNGFL, and DBLFL.

Figure 4.3-2 provides a high level functional description of the COVRGE program; a brief explanation of the functions in the figure follows:

- Computation Control

These subroutines control the computations for solving the single and double fault models; the details of the computational sequence are given in Section 4.3-2. Figures 4.3-3 and 4.3-4 illustrate the control structure of subroutines SNGFLT and DBLFLT with call trees; from a control point of view the two subroutines are quite similar.

- Single and Double Fault Functions

The subroutines in this group compute the basic distribution and survival function for the coverage model and the elementary functions used in the solution of the coverage equations; the function definitions are given in Section 4.3.3.

- Numerical Integration

The subroutines in this group are used to numerically compute the integral (over a time interval) of a function. This kind of calculation is required in the coverage model solution and for the evaluation of the moments of the coverage functions. The numerical methods used in these subroutines are discussed detail in Section 4.4.

- Numerical Convolution

The subroutines in this group are used to numerically compute the convolution of two functions and to solve Volterra integral equations of the second kind. This kind of calculation is required in the coverage solution. These are the most crucial numerical subroutines in the COVRGE program; the numerical methods used are discussed in detail in Section 4.4.

- Support Functions

These are "library" type subroutines which are used by all the other subroutines in the program for very basic operations or calculations.

Figure 4.3-5 illustrates the data structure used in the COVRGE program to store all functions of time that are computed during the course of the solution of the coverage model; this data structure will be referred to as a CARE-III, Type A function array. The figure shows that the discrete time array has a special structure: the step size only increases,

monotonically by factors of two. This data structure reflects the expectation, on the part of the CARE-III developers, that all functions of interest rapidly decay to zero or a steady-state value. The impact of this data structure on the performance of the numerical software in the COVRG program will be discussed in Section 4.4.

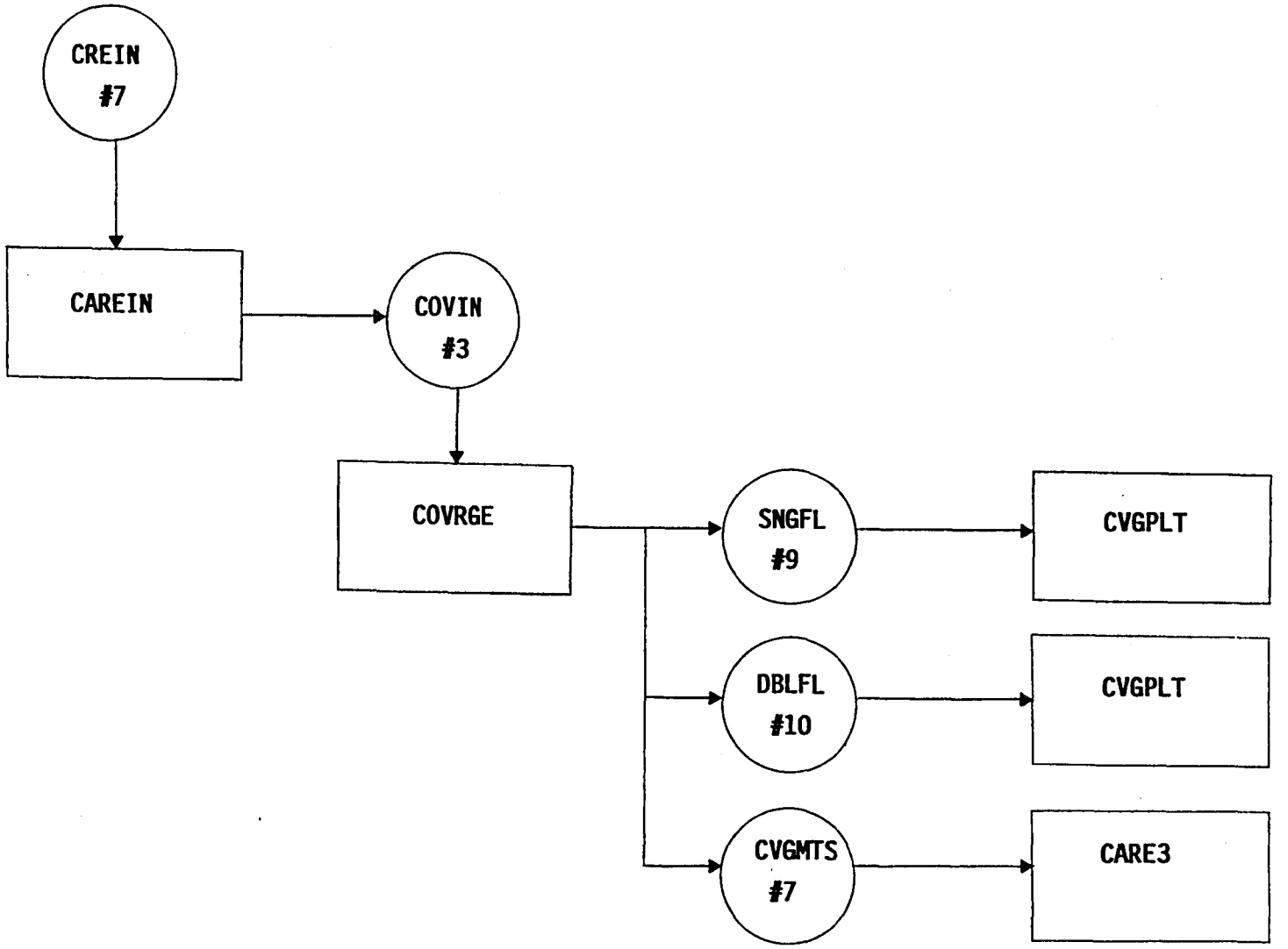


FIGURE 4.3-1 Data Flow for COVRGE Program

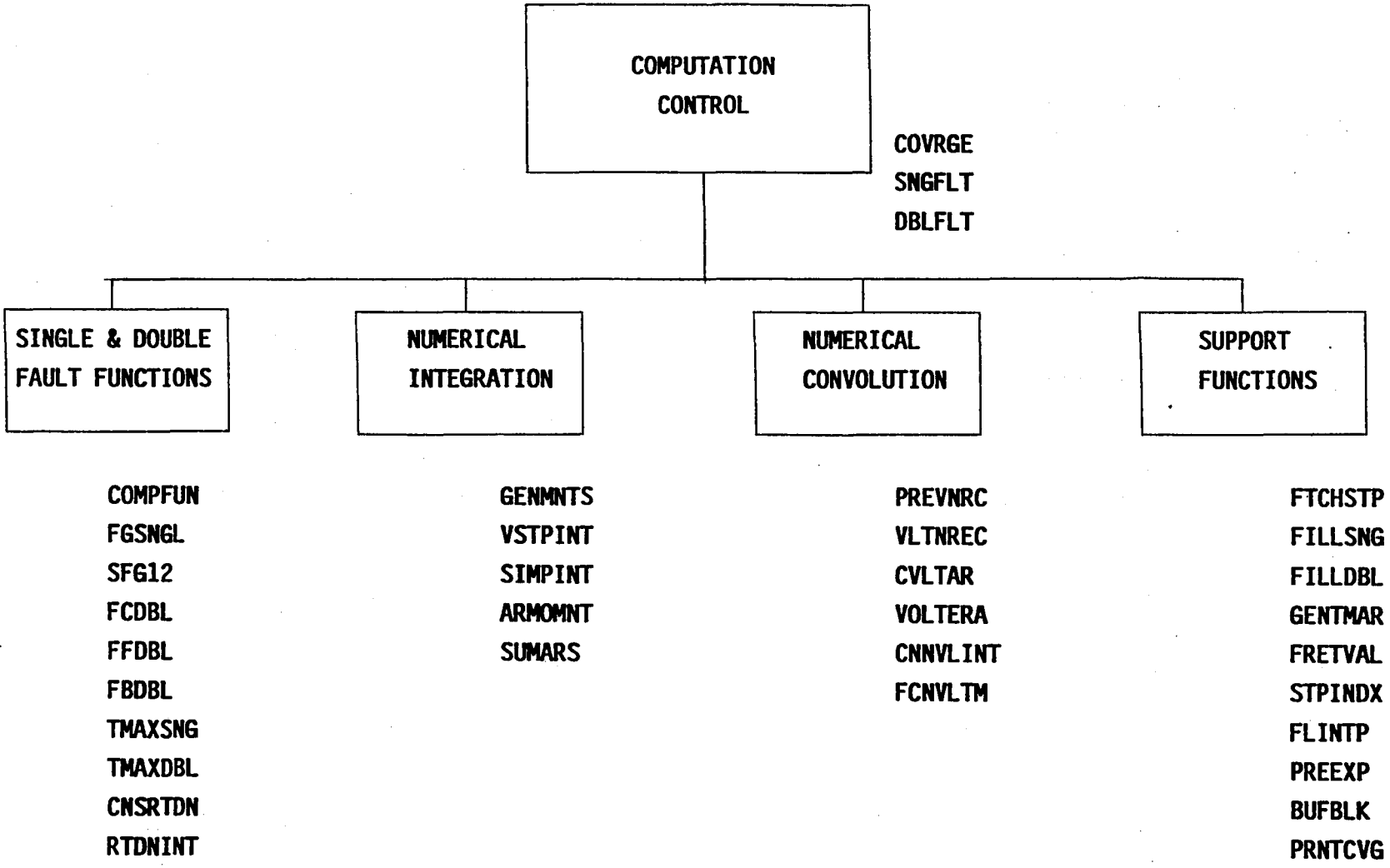


FIGURE 4.3-2 Functional Structure of COVRGE Program

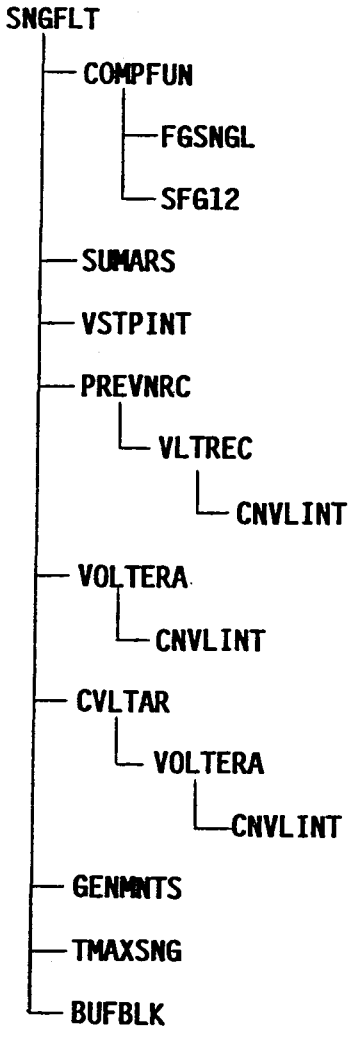


FIGURE 4.3-3 SNGFLT Call Tree

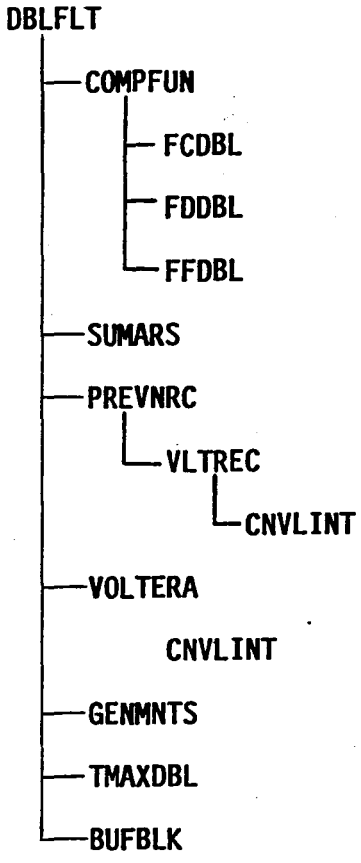
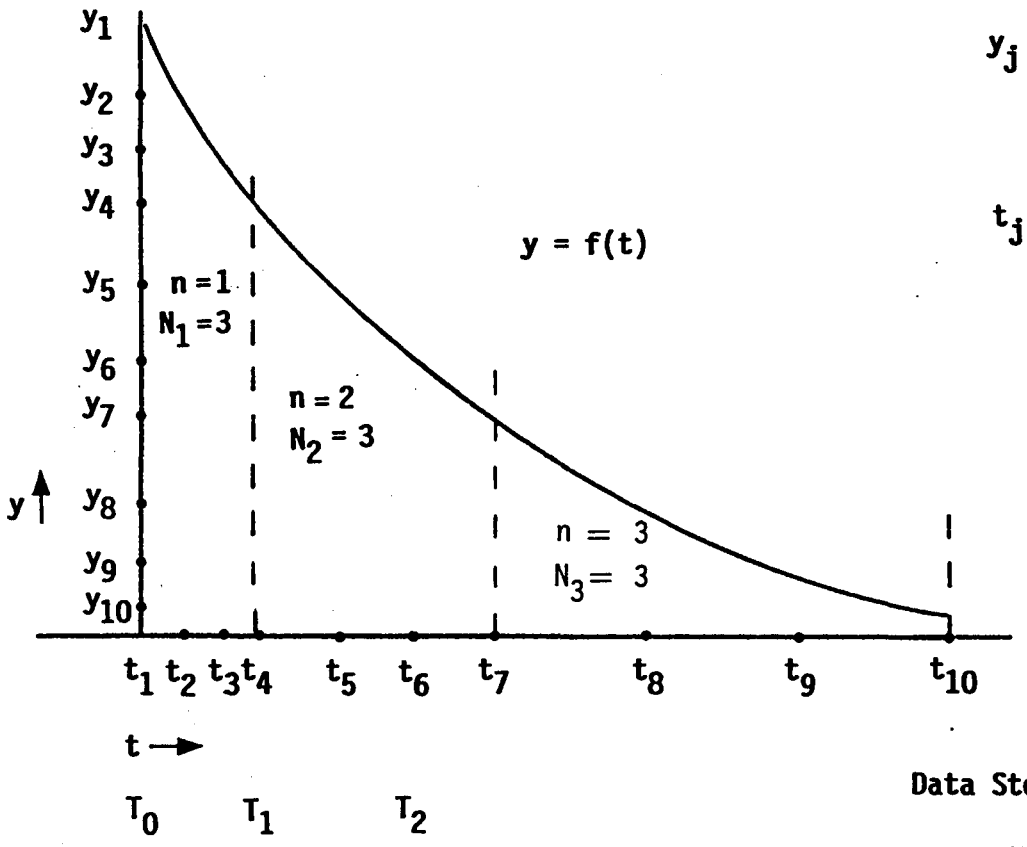


FIGURE 4.3-4 DBLFLT Call Tree



Function:

$$y = f(t)$$

Discrete Approximation:

$$y_j = f(t_j) : \; j = 1, 2, \ldots, j_{max}$$

$$t_j = \begin{cases} 0 & : j = 1 \\ k2^n \Delta t + \sum_{m=1}^{n-1} N_m \, 2^{m-1} \Delta t & : j = 1 + \sum_{m=1}^{n-1} N_m + k \end{cases}$$

where

$$1 \le n \le N_{max}$$

$$1 \le k \le N_n$$

Data Stored:

$$y_j, \Delta t, N_n$$

FIGURE 4.3-5 CARE-III Type-A Function Array

4.3.2 Outline of Calculations

The evaluation of the output coverage functions is performed by subroutines SNGFLT and DBLFLT under the control of the main program COVRGE. For each fault type, SNGFLT performs a series of calculations to evaluate all the equations of the single fault coverage model listed in Table 4.2-2. Moments of the functions p_{DP}, P_L, p_F, P_B and $P_{\overline{B}}$ are output by SNGFLT for later use in the reliability calculation. Similarly, for each pair of fault types, DBLFLT performs a series of calculations to evaluate all the equations of the double fault coverage model listed in Table 4.2-4. Moments of the function p_{DF} are output by DBLFLT for later use in the reliability calculation.

The calculations done in SNGFLT and DBLFLT are outlined in Tables 4.3-1 and 2, respectively. The arrays named as inputs or outputs for each calculation are CARE-III, Type A function arrays (see Figure 4.3-5). The calculation types are defined as follows.

- Function Evaluation (G or F)

The single fault functions, G_1 to G_{12}, are evaluated by subroutines FGSNGL and SFG12 and the double fault functions, F_{C1}, F_{C2}, F_{F1}, F_{F2}, F_{B1}, F_{B2}, are evaluated by subroutines FCDBL, FFDBL and FBDBL. In each case the function is evaluated under the control of subroutine COMPFUN which stores the function values as a CARE-III, Type A function array.

- Summation (SUM)

The sum of two functions is computed by subroutine SUMARS and stored as a CARE-III, Type A function array. SUMARS expects the two input functions to be stored as CARE-III, Type A function arrays.

- Integration (INT)

The integral of a function over a time interval is computed by subroutine VSTPINT and stored as a CARE-III, Type A function array. VSTPINT expects the input function to be stored as a CARE-III, Type A function array.

- Convolution (CNV)

The convolution of two functions is computed by subroutine PREVNRC (and VLTNREC) and stored as a CARE-III, Type A function array. PREVNRC expects the two input functions to be stored as CARE-III, Type A function arrays.

- Volterra Integral Equation (VIE)

The solution of a Volterra integral equation defined by an input function and a kernel function is computated by subroutines VOLTERA or CVLTAR and stored as a CARE-III, Type a function array. VOLTERA and CVLTAR expect the input and kernel functions to be stored as CARE-III, Type A function arrays.

The numerical procedures for computing sums, integrals, convolutions and solving Volterra integral equations are discussed in detail in Section 4.4.

TABLE 4.3-1

SINGLE FAULT CALCULATIONS

	Function	Calculation Type	Subroutine	Input Arrays	Output Arrays
1.	$\emptyset$	G_1	FGSNGL-1	-	FEEAR
2.	P_a	G_2	FSGNGL-2	-	GAR
		VIE	VOLTERA	GAR,FEEAR	PA
3.	P_b	VIE	VOLTERA	FEEAR,FEEAR	PB1
4.	P_e	G_3	FGSNGL-3	-	GAR
		G_4	FGSNGL-4	-	PERR
		CNV	PREVNRC	O.,PEER,GAR	FAR
		VIE	VOLTERA	FAR,FEEAR	PEER
5.	p_e	VIE		GAR,FEEAR	PEAR
6.	$p_{\overline{e}}$	G_5	FGSNGL-5		GAR
		VIE	VOLTERA	GAR,FEEAR	PNEAR
7.	p_f	G_6	FGSNGL-6		GAR
		CNV	PREVNRC	O.,GAR,PEAR	PFLD
8.	ψ_A	G_7	FGSNGL-7	-	GAR
		CNV	PREVNRC	PNEAR,GAR,PEAR	PSIA
9.	ψ_B	G_8	FGSNGL-8	-	GAR
		CNV	PREVNRC	O.,GAR,PEAR	PSIB
10.	X_B	G_9	FGSNGL-9	-	GAR
		CNV	PREVNRC	O.,GAR,PSIB	PB2

TABLE 4.3-1 (Continued)

	Function	Calculation Type	Subroutine	Input Arrays	Output Arrays
11.	P_{dp}	INT	VSTPINT	PSIA	GAR
		INT	VSTPINT	PSIB	FAR
		SUM	SUMARS	PSIA,PSIB	PDP
12.	$(1.-P_A)\psi_A+(1.-P_B)\beta\psi_B$	SUM	SUMARS	PSIA,PSB2	FEEAR
13.	ϵ	INT	VSTPINT	FEEAR	FAR
14.	P_{DP}	VIE	CVLTAR	PDP	PDP
15.	P_A	VIE	CVLTAR	PA	PA
16.	P_{B1}	VIE	CVLTAR	PB1	PB1
17.	P_{B2}	VIE	CVLTAR	PB2	PB2
18.	P_E	VIE	CVLTAR	PERR	PERR
19.	P_F	VIE	CVLTAR	PFLD	PFLD
20.	P_B	SUM	SUMARS	PB1,PB2	PBNG
21.	$P_{\bar{B}}$	SUM	SUMARS	PA,PERR	PNBNG
22.	P_L	SUM	SUMARS	PBNG,PNBNG	PLAT

TABLE 4.3-2

DOUBLE FAULT CALCULATIONS

	Function	Calculation Type	Subroutine	Input Arrays	Output Arrays
1.	C_1	F_{C1}	FCDBL-1	-	C1AR
2.	C_2	F_{C2}	FCDBL-2	-	C2AR
3.	f_1	F_{F1}	FFDBL-1	-	F1AR
4.	f_2	F_{F2}	FFDBL-2	-	F2AR
5.	b_1	F_{B1}	FBDBL-1	-	B1AR
6.	b_2	F_{B2}	FBDBL-2	-	B2AR
7.	C_4	CNV	PREVNRC	0.,C1AR,B1AR	XB1INTG
		CNV	PREVNRC	0.,C2AR,B2AR	XB2INTG
		SUM	SUMARS	XB1INTG,XB2INTG	C4AR
8.	C_3	CNV	PREVNRC	0.,F1AR,B1AR	XB1INTG
		CNV	PREVNRC	0.,F1AR,B2AR	XB1INTG
		SUM	SUMARS	XB1INTG,XB2INTG	C3AR
9.	P_3	VIE	VOLTERA	F1AR,C3AR	P3AR
10.	P_{DF}	VIE	VOLTERA	C1AR,C4AR	PDFAR

4.3.3 Basic Coverage Functions

The basic coverage functions are the distribution and survival functions for the various fault types and the elementary functions used in the evaluation of the equations for the coverage model (see Section 4.2). The distribution and survival functions are computed in subroutines CNSRTDN and RTDNINT, respectively. The single fault functions, G_1 to G_{12} are computed in subroutines FGSNGL and SFG12 and the double fault functions F_{C_1}, F_{C_2}, F_{F_1}, F_{F_2}, F_{B_1}, F_{B_2} are computed in subroutines FCDBL, FFDBL and FBDBL.

For a fault type, with parameters α, β, δ, ρ, ϵ, δ_F, ρ_F, and ϵ_F (see Section 4.2), the distribution and survival functions are computed as follows.

- Active-to-Benign Transition:

$$\alpha(t) = \alpha e^{-\alpha t} \qquad 4.3\text{-}1$$

$$a(t) = e^{-\alpha t} \qquad 4.3\text{-}2$$

- Benign-to-Active Transition:

$$\beta(t) = \beta e^{-\beta t} \qquad 4.3\text{-}3$$

$$b(t) = e^{-\beta t} \qquad 4.3\text{-}4$$

- Fault Detection:

$$\delta(t) = \begin{cases} \delta e^{-\delta t} & ; \ \delta_F = .F. \\ \delta & ; \ \delta_F = .T. \ , \ 0. \leq t \leq 1./\delta \\ 0. & ; \ \delta_F = .T. \ , \ t > 1./\delta \end{cases} \qquad 4.3\text{-}5$$

$$d(t) = \begin{cases} e^{-\delta t} & ; \ \delta_F = .F. \\ 1.-t\delta & ; \ \delta_F = .T. \ , \ 0. \leq t \leq 1./\delta \\ 0. & ; \ \delta_F = .T. \ , \ t > 1./\delta \end{cases} \qquad 4.3\text{-}6$$

- Error Generation:

$$\rho(t) = \begin{cases} \rho e^{-\rho t} & ; \ \rho_F = .F. \\ \rho & ; \ \rho_F = .T. \ , \ 0. \leq t \leq 1./\rho \\ 0. & ; \ \rho_F = .T. \ , \ t > 1./\rho \end{cases} \qquad 4.3\text{-}7$$

$$r(t) = \begin{cases} e^{-\rho t} & ; \ \rho_F = .F. \\ 1. - \rho t & ; \ \rho_F = .T. \ , \ 0. \leq t \leq 1./\rho \\ 0. & ; \ \rho_F = .T. \ , \ t > 1./\rho \end{cases} \qquad 4.3\text{-}8$$

- Error Propagation:

$$\epsilon(t) = \begin{cases} \epsilon e^{-\epsilon t} & ; \ \epsilon_F = .F. \\ \epsilon & ; \ \epsilon_F = .T. \ , \ 0. \leq t \leq 1./\epsilon \\ 0. & ; \ \epsilon_F = .T. \ , \ t \geq 1./\epsilon \end{cases} \qquad 4.3\text{-}9$$

$$e(t) = \begin{cases} e^{-\epsilon t} & ; \ \epsilon_F = .F. \\ 1. - \epsilon t & ; \ \epsilon_F = .T. \ , \ 0. \leq t \leq 1./\epsilon \\ 0. & ; \ \epsilon_F = .T. \ , \ t > 1./\epsilon \end{cases} \qquad 4.3\text{-}10$$

For a fault type, with parameters α, β, δ, ρ, ϵ, δ_F, ρ_F, ϵ_F (see Section 4.2), the single fault function G_1 to G_{12} are computed as follows:

- G_1 Function:

$$G_1(t) = \phi = G_{10}(t), \text{ with } \beta - \alpha \text{ in place of } \beta \qquad 4.3\text{-}11$$

- G_2 Function:

$$G_2(t) = a(t)\, r(t)\, d(t), \qquad 4.3\text{-}12$$

- G_3 Function:

$$G_3(t) = a(t)\, \rho(t)\, d(t), \qquad 4.3\text{-}13$$

- G_4 Function:

$$G_4(t) = e(t), \qquad 4.3\text{-}14$$

- G_5 Function:

$$G_5(t) = a(t)\, r(t)\, \delta(t), \qquad 4.3\text{-}15$$

- G_6 Function:

$$G_6(t) = \epsilon(t), \qquad 4.3\text{-}16$$

- G_7 Function:

$$G_7(t) = \begin{cases} \epsilon(t) & ; \alpha + \beta = 0. \\ \epsilon(t) \cdot \dfrac{\alpha a(t)b(t) + \beta}{\alpha + \beta} & ; \alpha + \beta \neq 0. \end{cases} \qquad 4.3\text{-}17$$

- G_8 Function:

$$G_8(t) = \epsilon(t)\,(1.-a(t)b(t)) \qquad 4.3\text{-}18$$

- G_9 Function:

$$G_9(t) = b(t) \qquad 4.3\text{-}19$$

- G_{10} Function:

$$G_{10}(t) = \alpha e^{-\alpha t} \int_0^t e^{-\beta(t-u)} d(u) r(u) du$$

- Case 1: $\alpha = 0.$

$$G_{10} = 0.$$

- Case 2: $\alpha > 0.$, $\delta_F = .F.$, $\rho_F = .F.$ 4.3-20

$$G_{10} = \begin{cases} \alpha a(t) t b(t) & : \rho + \delta = \beta \\ \alpha a(t) \dfrac{b(t) - r(t)d(t)}{\rho + \delta - \beta} & : \rho + \delta \neq \beta \end{cases} \qquad 4.3\text{-}21$$

- Case 3: $\alpha > 0.$, $\delta_F = .T.$, $\rho_F = .F.$ and $\rho - \beta \neq 0.$

$$t_o = \min(t, 1./\delta) \qquad 4.3\text{-}22$$

$$\text{exp 1} = e^{-(\alpha+\beta)t} \qquad 4.3\text{-}23$$

$$\text{exp 2} = e^{-(\alpha+\beta)t}\, e^{-(\rho-\beta)t_o} \qquad 4.3\text{-}24$$

$$\text{term 1} = (1./(\rho-\beta))\ (\text{exp 1} - \text{exp 2}) \qquad 4.3\text{-}25$$

$$\text{term 2} = (\delta / (\rho-\beta)^2)\ (\text{exp 1} - \text{exp 2}\ (1. + (\rho-\beta)\ t_o)) \qquad 4.3\text{-}26$$

$$G_{10}(t) = \alpha\ (\text{term 1} - \text{term 2}) \qquad 4.3\text{-}27$$

- Case 3b: $\alpha \geq 0.$, δ_F = .T., ρ_F = .F. and $\rho - \beta = 0.$

$$t_o = \min(t, 1./\delta) \qquad 4.3\text{-}28$$

$$G_{10}(t) = \alpha e^{-(\alpha+\beta)t}\ (t_o - 1/2\ \delta\ t_o^2) \qquad 4.3\text{-}29$$

- Case 4a: $\alpha > 0.$, δ_F = .F., δ_T = .T. and $\delta - \beta \neq 0.$

$$t_o = \min(t, 1./\rho) \qquad 4.3\text{-}30$$

$$\text{exp}\lambda 1 = e^{-(\alpha+\beta)t} \qquad 4.3\text{-}31$$

$$\text{exp 2} = e^{-(\alpha+\beta)t}\, e^{-(\delta-\beta)t_o} \qquad 4.3\text{-}32$$

$$\text{term 1} = (1./(\delta-\beta))\ (\text{exp 1} - \text{exp 2}) \qquad 4.3\text{-}33$$

$$\text{term 2} = (\rho/(\delta-\beta)^2)\ (\text{exp 1} - \text{exp 2}\ (1. + (\delta-\beta)t_o) \qquad 4.3\text{-}34$$

$$G_{10}(t) = \alpha \text{ (term 1 - term 2)} \qquad 4.3\text{-}35$$

- Case 4b: $\alpha \geq 0.$, $\delta_F = .F.$, $\delta_T = .T.$ and $\delta - \beta = 0.$

$$t_o = \min (t, 1./\rho) \qquad 4.3\text{-}36$$

$$G_{10}(t) = \alpha e^{-(\alpha+\beta)t} (t_o - 1/2\ \rho\ t_o^2) \qquad 4.3\text{-}37$$

- Case 5a: $\alpha > 0.$, $\delta_F = .T.$, $\rho_F = .T.$ and $\beta \neq 0.$

$$t_o = \min (t, 1./\delta, 1./\rho) \qquad 4.3\text{-}38$$

$$\text{exp 1} = e^{-(\alpha+\beta)t}\ e^{\beta t_o} \qquad 4.3\text{-}39$$

$$\text{exp 2} = e^{-(\alpha+\beta)t} \qquad 4.3\text{-}40$$

$$\text{term 1} = (1./\beta)\ (\text{exp 1} - \text{exp 2}) \qquad 4.3\text{-}41$$

$$\text{term 2} = ((\delta+\rho)/\beta^2)\ (\text{exp 1}\ (1.-\beta t_o) - \text{exp 2}) \qquad 4.3\text{-}42$$

$$\text{term 3} = (2\rho\delta/\beta^3)(\text{exp 2}\ (1 - \beta t_o + 1/2\ \beta^2 t_o^2) - \text{exp 1}) \qquad 4.3\text{-}43$$

$$G_{10}(t) = (\text{term 1} + \text{term 2} + \text{term 3}) \qquad 4.3\text{-}44$$

- Case 5b: $\alpha > 0.$, $\delta_F = .T.$, $\rho_F = .T.$ and $\beta = 0.$

$$t_o = \min (t, 1./\delta, 1./\rho) \qquad 4.3\text{-}45$$

$$G_{10}(t) = \alpha e^{-\alpha t}\left(t_o - 1/2\,(\delta+\rho)\,t_o^2 + 1/3\;\delta\rho\,t_o^3\right) \qquad 4.3\text{-}46$$

- G_{11} Function:

$$G_{11}(t) = \frac{\alpha}{\alpha+\beta}\,a(t)\,b(t) \qquad 4.3\text{-}47$$

- G_{12} Function:

$$G_{12}(t) = \frac{\beta}{\alpha+\beta}\left(G_1(t) - G_{10}(t)\right) + G_2(t) \qquad 4.3\text{-}48$$

For a pair of fault types, with parameters α_i, β_i, δ_i, ϵ_i; and α_j, β_j, ρ_j, ϵ_j; (see Section 4.2), the double fault function F_{C_1}, F_{C_2}, F_{F_1}, F_{F_2}, F_{B_1}, F_{B_2} are computed as follows.

- F_{C_1} Function:

$$\begin{aligned} F_{C_1}(t) &= \beta_i(t)d_j(t)r_j(t)a_j(t) \\ &\quad + (1.- P_{A_j})b_i(t)\,\delta_j(t)r_j(t)a_j(t) \\ &\quad + b_i(t)d_j(t)\,\rho_j(t)a_j(t) \end{aligned} \qquad 4.3\text{-}49$$

- F_{C_2} Function:

$$\begin{aligned} F_{C_2}(t) &= \beta_j(t)d_i(t)r_i(t)a_i(t) \\ &\quad + (1- P_{A_i})b_j(t)\,\delta_j(t)r_i(t)a_i(t) \\ &\quad + b_j(t)d_i(t)\,\rho_i(t)a_i(t) \end{aligned} \qquad 4.3\text{-}50$$

- F_{F_1} Function:

$$F_{F_1}(t) = \alpha_i(t)b_j(t)d_i(t)r_i(t) \qquad 4.3\text{-}51$$

- F_{F_2} Function:

$$F_{F_2}(t) = \alpha_j(t)b_i(t)d_j(t)r_j(t) \qquad 4.3\text{-}52$$

- F_{B_1} Function:

$$F_{B_1}(t) = b_i(t)\beta_j(t) \qquad 4.3\text{-}53$$

- F_{B_2} Function:

$$F_{B_2}(t) = b_j(t)\beta_i(t) \qquad 4.3\text{-}54$$

BCS has identified a coding error in subroutine FGSNGL for the calculation of $G_{10}(t)$, Case 5a; in equation 4.3-42 the order of the terms exp 1 and exp 2 should be switched.

4.4 COMPUTATIONAL METHODS

In the previous section the implementation of the coverage model in the COVRGE program was outlined, now, the computational methods used to solve the model equations will be reviewed. The objective of this section is to document the numerical procedures that are implemented and present the BCS evaluation of these algorithms in the CARE-III environment. In the following sections the numerical procedures are first highlighted and then discussed in detail.

4.4.1 Overview of Algorithms

In this section each of the numerical procedures used in the COVRGE program is highlighted. The requirements for the algorithm and its implementation are discussed first, followed by a preview of the BCS analysis of the algorithm in the CARE-III context. In the succeeding sections a detailed description and analysis of each algorithm is provided.

- Numerical Sum

  The calculation of several of the functions in the single and double fault coverage models (p_{dp}, ϵ, p_B, $p_{\overline{B}}$, p_L, C_3 and C_4) require the calculation of the sum of two functions. The summation procedure uses linear interpolation and is implemented in subroutine SUMARS. It is designed to compute the sum:

  $$y(t) = C_1 y_1(t) + C_2 y_2(t), \qquad 4.4\text{-}1$$

  for input functions stored in CARE-III, Type A function arrays. The output function is stored in a CARE-III, Type A function array which has discrete time points selected by the numerical procedure (in subroutine SUMARS).

BCS has carefully reviewed SUMARS and the subroutines which it uses; no programming errors in the implementation of the summation calculation were detected by the review (with the exception of some minor questions; see Section 4.4.2). BCS has raised questions about several of the heuristics used in the summation calculation: step size control and zero detection; these questions are addressed in Sections 4.4.2 and 4.4.6.

- Numerical Integration

  The calculation of several of the functions in the single fault coverage model (p_{dp}, ϵ) require the calculation of the integral of a function. In addition the output of the COVRGE program is the moments (for p = 0,1,2) of the single and double fault functions, p_{DP}, P_L, p_F, P_B, $P_{\overline{B}}$ and p_{DF}. The numerical integration procedure is based on Simpson's Rule and is implemented in subroutines VSTPINT and SIMPINT. It is designed to compute the integrals:

$$y^P(t) = \int_0^t \tau^P f(\tau)d\tau, \quad P = 0,1,2 \qquad 4.4\text{-}2$$

  for input functions which are stored in CARE-III, Type A function arrays. The output moment is stored in a CARE-III, Type A function array which has the same discrete time points as the input function.

BCS has carefully reviewed VSTPINT, SIMPINT and all the subroutines which they use; no programming errors in the implementation of Simpson's Rule were detected by the review.

- Numerical Convolution

  The calculation of several of the functions in the single and double fault coverage models (P_e, p_f, Ψ_A, Ψ_B, X_B, C_4, C_3) requires the

calculation of the convolution of two functions. The numerical convolution procedure is based on the Trapezoidal Rule and is implemented in subroutines PREVNRC, VLTNREC and CNVLINT. It is designed to compute the convolution:

$$y(t) = f(t) + \beta \int_0^t \phi\ (t-\tau)\ g\ (\tau)\ d\tau, \qquad 4.4\text{-}3$$

for input functions which are stored in CARE-III, Type A function arrays. The output function is stored in a CARE-III, Type A function array which has discrete time points selected by the numerical procedure (in subroutine VLTNREC).

BCS has carefully reviewed PREVNRC, VLTNREC, CNVLINT and the subroutines which they use; no programming errors in the implementation of the convolution calculation were detected by the review (with the exception of some minor questions on CNVLINT; see Section 4.4.4). BCS has raised questions about several of the heuristics used in the convolution calculations: step size control, zero detection and constant value detection; these questions are addressed in Sections 4.4.4 and 4.4.6.

- Numerical Solution of Volterra Integral Equations

The solution of Volterra integral equations of the second kind is an essential calculation for the single and double fault coverage models; such solutions are required to compute P_a, P_e, p_e, P_{DP}, P_A, P_{B1}, P_{B2}, P_E, P_F, P_3, P_{DF}. The numerical solution procedure is based on the Trapezoidal Rule and is implemented in subroutines VOLTERA, CVLTAR and CNVLINT. It is designed to solve integral equations of the form:

$$y(t) = f(t) + \beta \int_0^t \phi\ (t-\tau)\ y\ (\tau)\ d\tau, \qquad 4.4\text{-}4$$

for input functions which are stored in CARE-III, Type A function arrays. The output function is stored in a CARE-III, Type A function array which has discrete time points selected by the numerical procedure (in subroutine VOLTERA). Subroutine VOLTERA solves equation 4.4-4 directly for y(t) and CVLTAR computes the solution indirectly by solving a Volterra integral equation for y(t)-f(t).

BCS has carefully reviewed VOLTERA, CVLTAR, CNVLINT and the subroutines which they use; no programming errors in the implementation of the Volterra integral equation solution algorithm were detected by the review. BCS has raised questions about several of the heuristics used in the solution algorithm: step size control, zero detection and constant value detection; these questions are addressed in Sections 4.4.5 and 4.4.6. In addition, BCS has raised the important question of the numerical stability of the solution algorithm; this question is discussed in more detail in Section 4.4.5.

4.4.2 Numerical Sum

The numerical procedure used for calculating the sum of functions uses linear interpolation and is implemented in subroutine SUMARS. The input functions must be stored in a CARE-III, Type A function array and the sum is computed for a selected set of discrete time points:

$$y_k = C_1\, y_1(s_k) + C_2\, y_2(s_k);\ k = 1,2,\ldots,k_{max}. \qquad 4.4\text{-}5$$

The discrete time points for the sum function are automatically selected by SUMARS and the sum function is stored as a CARE-III, Type A function array.

The sum is computed with a step-by-step procedure that is initiated by setting:

$$s_1 = 0., \qquad 4.4\text{-}6$$

$$y_1 = C_1 y_1(s_1) + C_2 y_2(s1). \qquad 4.4\text{-}7$$

The k^{th} step consists of selecting a step size Δt_k for the step and then computing the sum at:

$$s_{k+1} = s_k + \Delta t_k, \qquad 4.4\text{-}8$$

$$y_{k+1} = C_1 y_1(s_{k+1}) + C_2 y_2(s_{k+1}) \qquad 4.4\text{-}9$$

Linear interpolation is used to evaluate the y_1 and y_2 functions at the discrete time points indicated in the sum in equation 4.4-9.

The summation procedure is monitored by heuristic controls which determine when the sum function is zero or may be truncated, select the stepsize Δt_k and determine when the sum function cannot be obtained in the available space in a CARE-III function array; each of these controls is briefly described below.

The summation procedure is terminated after computing y_k if one of the following conditions is met:

- $y_{k-2} = y_{k-1} = y_k = 0.$,
- $y_k \leq$ TRUNC (default value = .0001) and $s_k >$ maximum time for y_1 and y_2 functions,
- STDYFLG is set true and $s_k \geq$ FT (the flight time in hours),
- STDYFLG is set true and the maximum number of step doublings have been made.

The stepsize for the summation procedure is doubled after computing y_k if the following condition is met:

- $|y_k - y_{k-1}| / \max(|y_{k-2}|, |y_{k-1}|, |y_k|) \leq$ ZERODF.

The variable ZERODF (default value = .05), used to control the stepsize heuristic, is controlled to obtain the sum function in the available space as follows:

- ZERODF = ZERODF - DIFCHNG
  when the maximum number of step doublings is exceeded,

- ZERODF = ZERODF + DIFCHNG
  when the maximum number of function values is exceeded.

In both these cases the entire sum function is recomputed one more time; if either re-occurs an error message is displayed and the COVRGE program is terminated.

4.4.3 Numerical Integration

The numerical integration procedure used in the COVRGE program is based on Simpson's Rule and is implemented in subroutines VSTPINT and SIMPINT. The input function must be stored in a CARE-III Type A function array and the p^{th} moment of the function is evaluated at the same discrete time points as the input function:

$$y_j = f(t_j) \quad : \quad j = 1,2,\ldots,j_{max} \qquad 4.4\text{-}10$$

$$Y_j^p = \int_0^{t_j} \tau^p f(\tau) d\tau \quad : \quad j = 1,2,\ldots,j_{max}, p = 0,1,2 \qquad 4.4\text{-}11$$

Subroutine VSTPINT computes:

$$Y_j^p = \sum_{m=1}^{n-1} \int_{t_{j_m}}^{t_{j_{m+1}}} \tau^p f(\tau)d\tau + \int_{t_{j_n}}^{t_j} \tau^p f(\tau)d\tau \quad , \; j = 1,2,\ldots,j_{max} \qquad 4.4\text{-}12$$

where (refer to Figure 4.3-5)

$$j = 1 + k + \sum_{m=1}^{n-1} N_m \quad , \; 1 \leqslant k \leqslant N_n, \; 1 \leqslant n \leqslant N_{max} \qquad 4.4\text{-}13$$

$$j_n = \begin{cases} 1 & : n = 1 \\ 1 + \sum_{m=1}^{n-1} N_m & : n = 2,3\ldots,N_{max} \end{cases} \qquad 4.4\text{-}14$$

and the integrals (over the fixed stepsize intervals) are computed in subroutine SIMPINT via Simpson's Rule. For efficiency, the values of the integrals:

$$I_m^P = \int_{t_{j_m}}^{t_{j_{m+1}}} \tau^P f(\tau)d\tau \; : m = 1,2,\text{---},N_{max}\text{-}1, \qquad 4.4\text{-}15$$

are saved by VSTPINT and Y_j^P is computed as follows:

$$Y_j^P = \sum_{m=1}^{n-1} I_m^P + \int_{t_{j_n}}^{t_j} \tau^P f(\tau)d\tau \; . \qquad 4.4\text{-}16$$

Subroutine SIMPINT computes the integrals:

$$I_m^P(t_j) = \int_{t_{j_m}}^{t_j} \tau^P f(\tau)\,d\tau, \qquad 4.4\text{-}17$$

where the index j is defined by VSTPINT as follows:

$$j_m \leqslant j \leqslant j_{m+1} \qquad 4.4\text{-}18$$

The integrals are evalued by Simpson's Rule:

- Case 1: $j = j_m$

$$I_m^P(t_j) = 0. \qquad 4.4\text{-}19$$

- Case 2: $j - j_m = 1$

$$I_m^P(t_j) = \frac{2^{m-1}\Delta t}{6}\left(y_{j-1}t_{j-1}^P + 4\left(\frac{y_{j-1}+y_j}{2}\right)\left(\frac{t_{j-1}+t_j}{2}\right)^P + y_j\, t_j^P\right) \qquad 4.4\text{-}20$$

- Case 3: $j-j_m > 1$, $j-j_m$ even

$$I_m^P(t_j) = \sum_{i=j_m}^{j-2}{}^{*} \frac{2^{m-1}\Delta t}{3}\left(y_i t_i^P + 4y_{i+1}t_{i+1}^P + y_{i+2}\, t_{i+2}^P\right) \qquad 4.4\text{-}21$$

- Case 4: $j-j_m > 1$, $j-j_m$ odd

$$I_m^P(t_j) = \sum_{i=j_{m-1}}^{j-3}{}^{*} \frac{2^{m-1}\Delta t}{3}\left(y_i t_i^P + 4y_{i+1}\, t_{i+1}^P + y_{i+2} t_{i+2}^P\right)$$

$$+ \frac{3}{8}\left(y_{j-3}\, t_{j-3}^P + 3y_{j-2}\, t_{j-2}^P + 3\, y_{j-1}\, t_{j-1}^P + y_j\, t_j^P\right) \qquad 4.4\text{-}22$$

* Increment i by 2's

4.4.4 Numerical Convolution

The numerical convolution procedure used in the COVRGE program approximates the convolution integral with a discrete sum based on the Trapezoidal Rule and is implemented in subroutines PREVNRC, VLTNREC and CNVLINT. The input functions must be stored in a CARE-III, Type A function array and the convolution is computed for a selected set of discrete time points:

$$y_k = f(s_k) + \beta \int_0^{s_k} \phi(s_k - \tau) g(\tau) d\tau \quad ; \; k=1,2, \ldots, k_{max}. \qquad 4.4\text{-}23$$

The discrete time points for the output function are automatically selected by VLTNREC and the output is stored as a CARE-III, Type A function array.

The convolution is computed with a step-by-step procedure that is initiated by setting:

$$s_1 = 0. \qquad 4.4\text{-}24$$

$$y_1 = f(s_1) \qquad 4.4\text{-}25$$

The k^{th} step consists of selecting a stepsize Δt_k for the step and then computing the convolution integral at

$$s_{k+1} = s_k + \Delta t_k. \qquad 4.4\text{-}26$$

Writing the discrete time points for the ϕ and g functions as the sets:

$$T_\phi = \left\{ t_j^\phi : 1 \leq j \leq j_{max}^\phi \right\}, \qquad 4.4\text{-}27$$

$$T_g = \left\{ t_j^g : 1 \leq j \leq j_{max}^g \right\}, \qquad 4.4\text{-}28$$

the discrete time points used for the approximation of the convolution integral at s_{k+1} are

$$T_k = T_\phi^k \cup T_g^k = \left\{ t_j : 1 \leq j \leq N \right\}, \qquad 4.4\text{-}29$$

where

$$T_\phi^k = \left\{ s_{k+1} - t_j^\phi : t_j^\phi \leq s_{k+1},\ 1 \leq j \leq j_{max}^\phi \right\}, \qquad 4.4\text{-}30$$

$$T_g^k = \left\{ t_j^g : t_j^g \leq s_{k+1},\ 1 \leq j \leq j_{max}^g \right\}, \qquad 4.4\text{-}31$$

and the t_j in T_k are numbered so that the t_j are increasing in size.

It is noted that:

$$t_1 = 0 \qquad 4.4\text{-}31$$

$$t_N = s_{k+1} \qquad 4.4\text{-}32$$

The value of y_{k+1} is computed by applying the Trapezoidal Rule to evaluate the convolution integral at s_{k+1} (in subroutine (CNVLINT):

$$y_{k+1} = f(t_n) + \beta \left[\tfrac{1}{2}\, t_2 \phi(t_N) g(t_1) + \tfrac{1}{2}(t_N - t_{N-1})\, \phi(t_1) g(t_N) + \sum_{n=2}^{N-1} \tfrac{1}{2}(t_{n+1} - t_{n-1})\, \phi(t_N - t_n) g(t_n) \right] \qquad 4.4\text{-}33$$

Linear interpolation is used to evaluate the f, ϕ and g functions at the discrete time points indicated in the sum in equation 4.4-33.

The convolution procedure is monitored by heuristic controls which determine when the convolution is zero or may be truncated, select the stepsizes Δt_k and determine when the convolution cannot be obtained in the available space in a CARE-III, Type A function array; each of these controls is briefly described below.

The convolution procedure is terminated after computing y_k if one of the following conditions is met:

- $y_k \leqslant 0.$,
- $y_k \leqslant$ TRUNC (default value = .0001) and maximum number of step doublings have been made,

- $|y_k - y_{k-2}| \,/\, \max(y_{k-2}, y_{k-1}, y_k) \leq$ STDYDIF

  (default value = .0005) and maximum number of step doublings have been made.

The stepsize for the convolution procedure is doubled after computing y_k if the following condition is met:

- $|y_k - y_{k-1}| \,/\, \max(y_{k-2}, y_{k-1}, y_k) \leq$ ZERODF.

The variable ZERODF (default value = .05), used to control the stepsize doubling heuristic, is controlled to obtain the convolution in the available space as follows:

- ZERODF = ZERODF - DIFCHNG;
  when the maximum number of step doublings is exceeded,

- ZERODF = ZERODF + DIFCHNG;
  when the maximum number of function values is exceeded.

In both these cases the entire convolution is recomputed one more time; if either case re-occurs an error message is displayed and the COVRGE program is terminated.

4.4.5 Numerical Solution of Volterra Integral Equation

The numerical procedure used in the COVRGE program to solve Volterra integral equations of the second kind is based on the numerical convolution procedure described in Section 4.4.3 and is implemented in subroutine VOLTERA and CNVLINT. The input functions must be stored in a CARE-III, Type A function array and the solution is computed for a selected set of discrete time points:

$$y_k = f(s_k) + \beta \int_0^{s_k} \phi(s_k - \tau) y(\tau) d\tau \; ; \; k=1,2,\ldots,k_{max}. \qquad 4.4\text{-}34$$

The discrete time points for the solution are automatically selected by VOLTERA and the solution is stored as a CARE-III, Type A function array.

The solution is computed with a step-by-step procedure that is initiated by setting:

$$s_1 = 0. \qquad 4.4\text{-}35$$

$$y_1 = f(s_1) \qquad 4.4\text{-}36$$

The k^{th} step consists of selecting a stepsize Δt_k for the step and then solving the integral equation at

$$s_{k+1} = s_k + \Delta t_k. \qquad 4.4\text{-}37$$

First the set of discrete time points needed for approximation of the convolution integral at s_{k+1} is selected in the manner outlined in Section 4.4.4:

$$T_k = \left\{ t_j : 1 \leqslant j \leqslant N \right\}, \qquad 4.4\text{-}38$$

and it is noted that:

$$t_1 = s_1 = 0. \qquad 4.4\text{-}39$$

$$t_M = s_k, \text{ for some } M < N \qquad 4.4\text{-}40$$

$$t_N = s_{k+1}. \qquad 4.4\text{-}41$$

For $o \leqslant t \leqslant t_M$, $y(t)$ can be estimated by linear interpolation from the known values, $y_1, y_2, .. y_k$ of y at the discrete times $s_1, s_2, .., s_k$. For $t_M < t \leqslant t_N$, $y(t)$ must be estimated by linear interpolation of the values of y at t_N and t_M:

$$y(t) = y(t_M)\frac{t_N - t}{t_N - t_M} + y(t_N)\frac{t - t_M}{t_N - t_M} \qquad 4.4\text{-}42$$

$$= y_k \frac{t_N - t}{t_N - t_M} + y_{k+1} \frac{t - t_M}{t_N - t_M} \quad . \qquad 4.4\text{-}43$$

The value of y_{k+1} is computed by applying the Trapezoidal Rule to evaluate the convolution integral at s_{k+1} (in subroutine CNVLINT):

$$y_{k+1} = f(t_N) + \beta \left[\tfrac{1}{2}\, t_2 \phi(t_N)\, y_1 + \tfrac{1}{2}\, (t_N - t_{N-1})\, \phi(t_1)\, y_{k+1} \right.$$

$$\left. + \sum_{n=2}^{N-1} \tfrac{1}{2}\, (t_{n+1} - t_{n-1})\, \phi(t_N - t_n)\, y\,(t_n) \right] \qquad 4.4\text{-}44$$

$$y_{k+1} = f(t_N) + \beta \left[\tfrac{1}{2}\, t_2\, \phi(t_N)\, y_1 + \tfrac{1}{2}\,(t_N - t_{N-1})\,\phi(t_1)\, y_{k+1} \right.$$

$$+ \sum_{n=2}^{M} \tfrac{1}{2}\,(t_{n+1} - t_{n-1})\,\phi(t_N - t_n)\, y(t_n)$$

$$\left. + \sum_{n=M+1}^{N-1} \tfrac{1}{2}\,(t_{n+1} - t_{n-1})\,\phi(t_N - t_n) \left(y_k \frac{t_N - t_n}{t_N - t_M} + y_{k+1} \frac{t_n - t_M}{t_n - t_M} \right) \right]$$

4.4-45

$$y_{k+1} = \frac{f(t_N) + \beta \left[\tfrac{1}{2}\, t_2\, \phi(t_N)\, y_1 + \sum_{n=2}^{M} \tfrac{1}{2}\,(t_{n+1} - t_{n-1})\,\phi(t_N - t_n)\, y(t_n) + y_k \sum_{n=M+1}^{N-1} \tfrac{1}{2}\,(t_{n+1} - t_{n-1})\,\phi(t_N - t_n) \frac{t_N - t_n}{t_N - t_M} \right]}{1 - \beta \left[\sum_{n=M+1}^{N-1} \tfrac{1}{2}\,(t_{n+1} - t_{n-1})\,\phi(t_N - t_n) \frac{t_n - t_M}{t_N - t_M} - \tfrac{1}{2}\,(t_N - t_{N-1})\,\phi(t_1) \right]}$$

4.4-46

Linear interpolation is used to evaluate the f, ϕ and y functions at the discrete time points indicated in equation 4.4-46.

The solution procedure is monitored by heuristic controls which determine when the solution is zero or may be truncated, select the stepsizes Δt_k and determine when a solution cannot be obtained in the available space in a CARE-III, Type A function array; each of these controls is briefly described below.

The solution procedure is terminated after computing y_k if one of the following conditions is met:

- $y_k \leq$ REALMIN (default value = 10^{-293}),
- $y_k \leq$ TRUNC (default value = .0001), and $s_k >$ maximum time for ϕ function,
- $y_k \leq$ TRUNC (default value = .0001), and maximum number of step doublings have been made,
- STDYFLG is set true and $s_k \geq$ FT (the flight time in hours),
- STDYFLG is set true and the maximum number of step doublings have been made.

The stepsize for the solution procedure is <u>doubled</u> after computing y_k if one of the following conditions is met:

- $|y_k - y_{k-2}| / \max(y_{k-2}, y_{k-1}, y_k) \leq$ ZERODF;
  The stepsize is doubled if y(t) has not had a relative maxima or minima in IHLDDUB steps. In addition the control flag STDYFLG is set to true if y(t) does not have a relative maxima or minima at s_k.
- $|y_k - y_{k-1}| / \max(y_{k-2}, y_{k-1}, y_k) \leq$ ZERODF;
  The stepsize is doubled if y(t) has not had a relative maxima or minima in IHLDDUB steps.

The variable ZERODF (default value = .05), used to control the stepsize doubling heuristic, is controlled to obtain the solution in the available space as follows:

- ZERODF = ZERODF - DIFCHNG;
  when the maximum number of step doublings is exceeded,

- ZERODF = ZERODF + DIFCHNG;
  when the maximum number of function values is exceeded and the last value of y is less than 1.,

- ZERODF = ZERODF - DIFCHNG;
  when the maximum number of function values is exceeded and the last value of y is greater than 1.

In both these cases the entire solution is recomputed one more time; if either case re-occurs an error message is displayed and the COVRGE program is terminated.

Section 5
REFERENCES

Bryant, L. A., and J. J. Stiffler (1982a), CARE III Phase II Report Maintenance Manual, NASA Contractor Report 165863.

Bryant, L. A., and J. J. Stiffler (1982b), CARE III Phase II Report User's Manual, NASA Contractor Report 165863.

Cinlar, E. (1975), Introduction to Stochastic Processes, Prentice Hall, Englewood Cliffs, N. J.

Feller, W. (1968), An Introduction to Probability Theory and Its Applications, John Wiley and Sons, New York.

Parzen, E. (1967), Stochastic Processes, Holden-Day, San Francisco.

Ross, S. M. (1970), Applied Probability Models With Optimization Applications, Holden-Day, San Francisco.

Stiffler, J. J., and L. A. Bryant (1982), CARE III Phase II Report - Mathematical Description, NASA Contractor Report 3566.

Stiffler, J. J., L. A. Bryant, L. Guccione (1979), CARE III Final Report, Phase I, Volume I, NASA Contractor Report 159122.

Stiffler, J. J., J. S. Neumann and L. A. Bryant (1982), CARE III Phase III Final Report - Test and Evaluation, NASA Contractor Report 3631.

CARE III References Not Specifically Cited in Text:

Stiffler, J. J., L. A. Bryant, L. Guccione (Nov. 1979), CARE III Final Report, Phase I, Volume II, NASA Contractor Report 159123.

Trivedi, K. S., and J. B. Clary, eds. (1980), Validation Methods Research for Fault-Tolerant Avionics and Control Systems Sub-Working Group Meeting - CARE III Peer Review, NASA Conference Publication 2167.

Trivedi, K. S., and R. M. Geist (1981), A Tutorial on the CARE III Approach to Reliability Modeling, NASA Contractor Report 3488.

A. APPENDIX

A.0 MARKOV AND SEMI-MARKOV PROCESSES

This appendix gives brief descriptions and some properties of Markov and Semi-Markov processes. References on these topics are Parzen (1967), Feller (1968), Ross (1970) and Cinlar (1975).

Consider a physical system that moves from one state to another with random sojourn times in between. Assume that the number of possible states is finite, and the paths that follow the state of the system are right continuous and piecewise constant:

i.e., if $X(0)$, $X(1)$, ... are the successive states visited; $0 = T(0) \leqslant T(1) \leqslant \ldots$ are the successive times of jump; and $Y(t)$ is the state of the system at the time t, then the system starts in state $X(0)$ at time $T(0)=0$, and remains there until time $T(1)$,

$$Y(t) = X(0) \quad \text{for} \quad T(0) \leqslant t < T(1);$$

at time $T(1)$ it jumps to state $X(1)$ and remains there until time $T(2)$,

$$Y(t) = X(1) \quad \text{for} \quad T(1) \leqslant t < T(2); \text{ and so on.}$$

In general

$$Y(t) = X(n) \quad \text{for} \quad T(n) \leqslant t < T(N+1).$$

Equivalently,

$$X(n) = Y(T(n)), \text{ and}$$
$$T(n) = \inf \left\{ t \geqslant T(n-1) \;/\; Y(t) \neq Y(T(n-1)) \right\}.$$

The processes $Y = \{Y_t : t \geq 0\}$ and $(X,T) = \{(X_n, T_n) : n \geq 0\}$ are "equivalent" in the sense that they give the same information.

The problem is to determine the state probabilities

$$P_{ij}(t) = P\left[Y(t) = j \mid Y(0) = i\right]$$

under some assumptions on the stochastic model (time homogeneity, independence from past history, distributions of sojourn times).

The following notation will be used

$$Q_{ij}(t) = P\left[X(n+1) = j \ , \ T(n+1) - T(n) \leq t \mid X(n) = i\right] \ ,$$

$$Q_{ij} = P\left[X(n+1) = j \mid X(n) = i\right] = Q_{ij}(\infty)$$

$$G_{ij}(t) = P\left[T(n+1) - T(n) \leq t \mid X(n) = i, X(n+1)=j\right] = Q_{ij}(t)/Q_{ij}.$$

A.1 MARKOV PROCESSES

Definition $Y = \{Y(t) : t \geq o\}$ is said to be a Markov process if

$$P\left[Y(t+s) = j \mid Y(u) : u \leq t\right] = P\left[Y(t+s) = j \mid Y(t)\right]$$

for all $t, s \geq o$. I.e., the future of the process is independent of its past provided that the present state is known.

If $P_{ij}(s,t)$ is used to denote the probability that the system is in state j at time t, given that it was in state i at time s $(s \leq t)$, then these satisfy the Chapman-Kolmogorov equations,

$$P_{ij}(s,t) = \sum_k P_{ik}(s,u) \, P_{kj}(u,t) \quad \text{for} \quad s \leq u \leq t \ .$$

Assumptions

(a) For every state i there is a non-negative continuous function $\lambda_i(t)$ such that

$$\left[1 - P_{ii}(t, t+h)\right]/h \text{ tends to } \lambda_i(t) \text{ as } h \text{ tends to } 0.$$

(b) For each pair of distinct states i, j, there corresponds transition probabilities $q_{ij}(t)$ such that

$$P_{ij}(t, t+h)/h \text{ tends to } \lambda_i(t)\, q_{ij}(t) \text{ as } h \text{ tends to } 0.$$

The functions $q_{ij}(t)$ are continuous in t, equal to zero when i=j, and for each fixed i add up to one.

The state probabilities are then evaluated using either the forward or backward equations:

(1) forward equations

$$\frac{d}{dt} P_{ij}(s,t) = -P_{ij}(s,t)\,\lambda_j(t) + \sum_{k\neq j} P_{ik}(s,t)\lambda_k(t)\, q_{kj}(t)$$

$$P_{ij}(s,t) = \delta_{ij} \exp\left\{-\left[\Lambda_j(t) - \Lambda_j(s)\right]\right\} + \sum_{k\neq j} \int_s^t P_{ik}(s,u)\,\lambda_k(u)\, q_{kj}(u) \exp\left\{-\left[\Lambda_j(t) - \Lambda_j(u)\right]\right\} du,$$

where $\Lambda_j(t) = \int_0^t \lambda_j(u)\, du.$

(2) backward equations

$$\frac{d}{ds} P_{ij}(s,t) = \lambda_i(s)\, P_{ij}(s,t) - \lambda_i(s) \sum_{k\neq i} q_{ik}(s)\, P_{kj}(s,t)$$

$$P_{ij}(s,t) = \delta_{ij} \exp\left\{-\left[\Lambda_i(t) - \Lambda_i(s)\right]\right\} + \sum_{k\neq i} \int_s^t \exp\left\{-\left[\Lambda_i(u) - \Lambda_i(s)\right]\right\} \lambda_i(u)\, q_{ik}(u)\, P_{kj}(u,t)\, du.$$

A.2 HOMOGENEOUS MARKOV PROCESSES

If in the definition of a Markov process it is also assumed that transition probabilities $P_{ij}(s,t)$ depend on the times s and t only through their difference t-s, then the process is said to be homogeneous and the term $P_{ij}(t-s)$ is used instead.

For this case the functions $_i$ and q_{ij} are independent of time and so will be written without reference to that parameter. Such processes satisfy the following properties:

$$(1)\quad P\Big[X(n+1) = j;\ T(n+1)-T(n) > t \,|\, X(o)\ldots X(n),\ T(o),\ldots,\ T(n)\Big] = q_{ij}\exp\left(-\lambda_i t\right) \quad \text{when } X(n) = i.$$

Where $q_{ij}\geqslant o$, $q_{ii} = o$, $\sum_j q_{ij}=1$ and $0\leqslant\lambda_i\leqslant\infty$

The state i is said to be absorbing, stable or instantaneous depending on whether $\lambda_i=o$, $o<\lambda_i<\infty$, or $\lambda_i=\infty$.

(2) $X = \{X(n) : n\geqslant o\}$ is a Markov chain with transition matrix $Q = [q_{ij}]$.

(3) The times between jumps are conditionally independent given the successive states being visited, and each sojourn time is exponentially distributed with parameter dependent on the state being visited, i.e.,

$$P\Big[T(1)-T(o)>u_1,\ \ldots,\ T(n)-T(n-1)>u_{n-1} \mid X(o)=i_o,\ \ldots,\ X(n)=i_n\Big] = \exp\left\{-\left[\lambda_{i_o}u_1 + \ldots + \lambda_{i_{n-1}}u_n\right]\right\}.$$

The state probabilities are evaluated using either the forward or backward equations. e.g., the forward equation becomes

$$P_{ij}(t) = \delta_{ij}\exp\left(-\lambda_j t\right) + \sum_{k\neq j}\int_o^t P_{ik}(u)\lambda_k\, q_{kj}\exp\left[-\lambda_j(t-u)\right]du.$$

A.3 SEMI-MARKOV PROCESSES

Definition: $Y = \{Y(t) : t \geq o\}$ is said to be a Semi-Markov process if for any s and any time of jump T (T=T(o) or T(1) or ...),

$$P\left[Y(T+s) = j \mid Y(u),\ u \leq T\right] = P\left[Y(T+s) = j \mid Y(T)\right].$$

i.e., the future and the past are conditionally independent given the present if the present is a time of transition.

Such a process has the following properties:

(1) $X = \{X(n) : n \geq o\}$ is a Markov Chain with transition probability matrix $Q = \left[q_{ij}\right]$.

(2) Sojourn times are conditionally independent but their distributions depend both on the state being visited and the next one. Also these distributions are arbitrary (not necessarily exponential). i.e.,

$$P\left[T(1) - T(o) \leq u_1, \ldots, T(n) - T(n-1) \leq u_n \mid X(o), \ldots, X(n)\right] =$$

$$= G(X(o), X(1), u_1) \ldots G(X(n-1), X(n), u_n).$$

Notation

$N_j(t)$ = number of transitions into state j in $(o,t]$;

$R_{ij}(t) = \delta_{ij} + E\left[N_j(t) \mid Y(o)=i\right]$ = expected number of visits to state j in $[o,t]$ given that the initial state is i;

$F_{ij}(t) = P\left[N_j(t) > o \mid Y(o)=i\right] =$ distribution of time between an entry to state i and the first next entry to j.

$$f_{ij} = \frac{d}{dt} F_{ij}(t) = \lim_{h \to o} \frac{1}{h} P\left[N_j(t+h)>o,\ N_j(t) = o \mid Y(o)=i\right] =$$

= Intensity of entry into state j at time t, given that the initial state is i.

Properties

$$(3)\quad R_{ij}(t) = \delta_{ij} + \sum_k \int_o^t Q_{ik}(ds)\, R_{kj}(t-s)$$

$$F_{ij}(t) = Q_{ij}(t) + \sum_{k \neq j} \int_o^t Q_{ik}(ds)\, F_{kj}(t-s)$$

$$(4)\quad P_{ij}(t) = \int_o^t R_{ij}(ds)\, h_j(t-s)$$

$$= \delta_{ij}\, h_i(t) + \sum_k \int_o^t Q_{ik}(ds)\, P_{kj}(t-s)$$

$$= \delta_{ij}\, h_i(t) + \int_o^t F_{ij}(ds)\, P_{jj}(t-s)$$

where

$$h_i(t) = 1 - \sum_k Q_{ik}(t) = P\left[T(n+1)-T(n)>t \mid X(n)=i\right] .$$

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-166096		

4. Title and Subtitle	5. Report Date
Review and Verification of CARE III Mathematical Model and Code: Interim Report	April 1983
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
D. M. Rose, R. E. Altschul, J. W. Manke and D. L. Nelson	BCS 40389
	10. Work Unit No.

9. Performing Organization Name and Address	11. Contract or Grant No.
Boeing Computer Services Company Energy Technology Applications Division Seattle, WA 98124	NAS1-16900

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
National Aeronautics and Space Administration Washington, DC 20546	Contractor Report Jan.-Nov. 1982
	14. Sponsoring Agency Code 505-34-13

15. Supplementary Notes

NASA Project Engineer: Salvatore J. Bavuso
NASA Langley Research Center
Hampton, VA 23665

16. Abstract

An independent verification of the CARE III mathematical model and computer code was conducted. The methodology of the verification and the results are reported in this document together with details of the CARE III mathematical model and the computer code.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Reliability modeling CARE III Fault coverage Fault models Fault-tolerant avionics	UNCLASSIFIED - UNLIMITED SUBJECT CATEGORY 59

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
UNCLASSIFIED	UNCLASSIFIED	185	A09

For sale by the National Technical Information Service, Springfield, Virginia 22161

End of Document